COMMODOREWELT

128 SPECIAL

DM 14,80-ÖS 124-SFR 14,80

1/89

CW-SPECIAL 128



SOFTWARE JAHRBUCH 1989

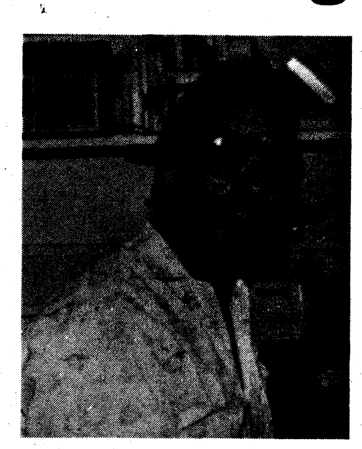
Die besten Anwender-Programme

Die schönsten Spiele

Hilfreiche Utilities

Das Super-Sonderheft für Ihren 128 PC

Guten Tag



Der C128 nimmt eine Ausnahmestellung unter den Acht-Bit-Computern ein. Mit sehr viel Vorschußlorbeeren startete er 1985 und wurde von COMMODORE als Nachfolger des C64 präsentiert. Ausgerüstet mit drei Betriebssystemen sollte er auch im professionellen Bereich den damals noch lückenhaften Markt füllen. Dabei wurden die C64-Umsteiger nicht vergessen. Ist doch der 64er-Modus ein fester Bestandteil des C128. Mit dem BASIC V7.0 im 128er-Modus wurde eine erweiterte und verbesserte Sprache realisiert, die dem Anwender erlaubte, besonders im Grafikbereich, professionelle Programme zu erstellen.
Last but not least wäre noch das dritte Betriebssystem CP/M zu nennen. Das vor drei Jahren noch zum Standard gehörende Betriebssystem sollte dem C128 den Einstieg in den professionellen Sektor ermöglichen. Mittlerweile wurde dieser Standard von den MS-DOS-Rechnern verdrängt. Auch COMMODORE stellte sich sehr schnell auf diesen Standard um und brachte mit der PC-Reihe MS-DOS-Rechner auf den Markt. Somit wurde der C128 immer mehr in die Rolle eines Außenseiters gedrängt.
Da sehr viele Software- und Hardwarefirmen nur für den C64 produzieren und den 128er-Modus links liegen lassen, sind mehr denn je die Programmier-Freaks und Hardwarebastler gefragt. Gibt es doch viele, die mit dem Angebot nicht zufrieden sind und daher allein im stillen Kämmerlein ihre eigenen Programme und Erweiterungen ersinnen. Für sie und für Sie daher unser Software-Jahrbuch.
Besonderen Augenmerk legten wir auf eine abwechslungsreiche Mischung. So finden Sie aus jeder Sparte des Computerns Beiträge und Listings. Wir führen Sie in die Welt der Grafik ein und geben Ihnen eine BASIC-Erweiterung in die Hand, in der Sie einige neue und nützliche Befehle finden, wie zum Beispiel MERGE und FIND. Die Musiker unter den 128er-Usern kommen auch nicht zu kurz. Mit dem Programm Musik-Genie werden sie schnell zum Komponisten. Eine nahezu professionelle Textverarbeitung im 60-Zeichen-Modus ist ebenfalls enthalten.
Leider nicht sehr verbreitet ist das Arbeiten unter CP/M. Was ja auch gar nicht so einfach ist, wenn Sie nur die Betriebssystem-Diskette CP/M 3.0 besitzen. Um eine Datei, ein File zu erstellen, brauchen Sie unbedingt ein Zusatzprogramm, wie es beispielsweise BASIC für das „normale" Betriebssystem des C128 ist. Das könnte ohne weiteres Turbo-Pascal, eine rasante Abwandlung dieser bekannten Programmiersprache, sein.
Mehr über das dritte Betriebssystem CP/M finden Sie auf den entsprechenden Seiten in diesem Heft. Ebenso sind wir dem Wunsch vieler Leser nachgekommen, eine Einführung in die Datenfernübertragung zu bringen. Im kleinen Dfü-Lexikon können Sie alle wichtigen Begriffe und deren Erklärungen nachlesen. Für alle Spieler halten wir die Programme Börsenspiel, Puker und Oilcompany bereit. Abgerundet wird das Jahrbuch mit Tips und Tricks sowie nützliche Peeks und Pokes.
Einige Hinweise noch zum Abtippen unserer Listings: Alle BASIC-Listings geben Sie bitte mit unserem Checksummer ein. Für die Maschinensprache-Listings verwenden Sie bitte den Checkmon. Alle Listings sind mit einer Checksumme versehen. Die in spitzen Klammern stehende Checksumme bitte nicht mit abtippen! Sie dient lediglich der Überprüfung, ob eine bestimmte Buchstabenfolge mit der auf dem Bildschirm angezeigten übereinstimmt.
Wem das Abtippen der Listings zu mühselig ist, der kann bequem und problemlos unseren Diskettenservice in Anspruch nehmen. Dazu beachten Sie bitte die Hinweise und den Bestell-Coupon auf Seite 64.
Ihr
Andreas Greil

COMMODOREWELT-SPECIAL 1/89

IMPRESSUM

64/128

C64/128 SPECIAL erscheint zweimonatlich in der CA-Verlags GmbH, einer Gesellschaft der Aktuell-Gruppe

VERANTWORTLICH FÜR DEN INHALT:
Andreas Greil

GESCHÄFTSFÜHRER:
Werner E. Seibt

REDAKTION UND STÄNDIGE MITARBEITER:
Peter Basch, Rosemarie Huber, Lothar Miedel, Michael Reppisch, Rudolf Schmid-Fabian, Torsten Seibt, Hermann Wellesen, Bernd Welte

ANZEIGENVERWALTUNG:
ADV-Mediendienste, Aindlingerstraße 17–19, 8900 Augsburg 1
Tel.: (0821)7904-227
Telekopierer: (0821)7904-243
Telex: adv 533502
Teletex: 821887

VERANTWORTLICH FÜR DEN ANZEIGENINHALT:
Brigitte Kostić

ANSCHRIFT FÜR ALLE VERANTWORTLICHEN:
Postfach 1161, 8044 Unterschleißheim,
Tel.: 089/1298011
Telex: 5214428 cav-d
Es gilt Preisliste Nr. 9 vom 1.6.1988
Media-Unterlagen bitte anfordern.

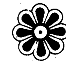
Printed in Germany

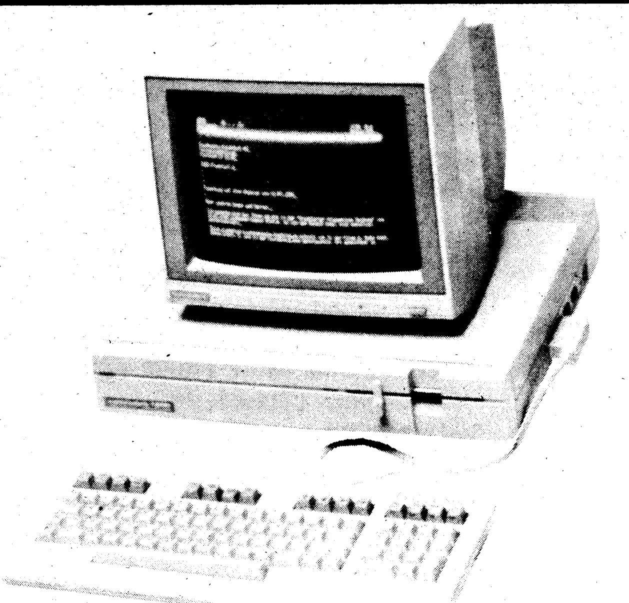
TEST

TIPS & TRICKS

SERVICE







LISTINGS

HALT

GRAFIK IN DER ANDEREN DIMENSION

Der C128 wird in der Grundkonfiguration mit einem Baustein ausgeliefert, dessen Fähigkeiten noch kaum einer breiten Öffentlichkeit bekannt wurden: wir meinen den VDC 8563, den Video-Display-Controller für den 80-Zeichen-Modus. Wenn Sie nun glauben, dieser Chip wäre einzig und allein nur dazu fähig, Ihren Eingabetext mit 80 Zeichen darzustellen, dann haben Sie ihn gewaltig unterschätzt.

Die Brüder Giger aus Solothurn in der Schweiz, Inhaber der Fa. Combo AG, gaben sich mit den spärlichen Informationen über diesen Chip aus dem Handbuch und die, die von zuständiger Seite zu erhalten waren, nicht zufrieden. Sind doch mit dem BASIC 7.0 im 40-Zeichen-Modus mit dem anderen Videobaustein VIC sehr schöne

BISLANG NUR GRAFIK IM 40-ZEICHEN-MODUS

Grafiken auf die einfachste Weise zu erstellen. Befehle wie GRAPHIC, DRAW, BOX CIRCLE haben bei der Arbeit mit dem 80-Zeichen-Chip keinerlei Wirkung. Es gibt zwischenzeitlich eine oder zwei Software Erweiterungen für den C128, die das auch ermöglichen, allerdings können sie in de Qualität der Bildschirmausgabe und in ihrer begrenzten Auflösung (640*200) den verwöhnten Grafik-Fan nicht recht zufriedenstellen.

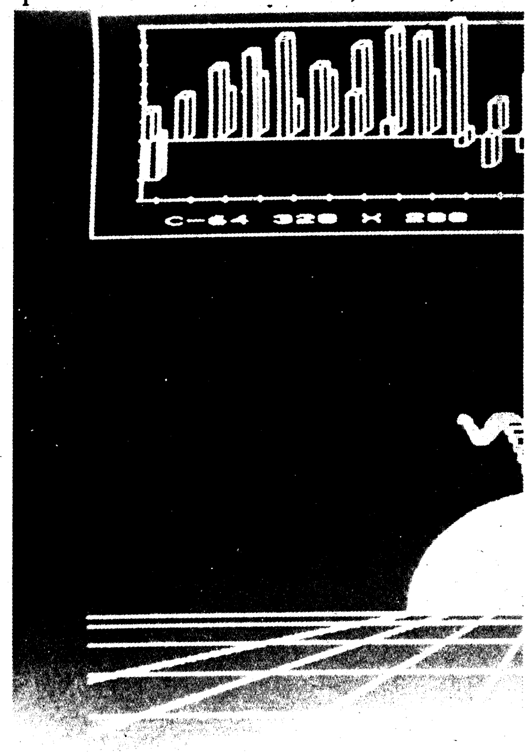
Daß es auch anders geht, beweist uns das kombinierte Soft-/Hardware-Paket „Graphic Booster 128" aus der Schweiz. (Augenblicklicher Preis: 174 Mark + MWSt.)

Es enthält eine Zusatz-Steckplatine mit genauer Bauanleitung und die notwendige Installations-Software. Etwas Zeit, Geduld und Konzentration (aber keinen Lötkolben!) müssen Sie beim Einbau der Steckkarte schon aufwenden, wobei der Einbau bei der „Normal"-Ausgabe des C128 ungleich leichter von statten geht als bei der 128-D-Ausfüh-

EINBAU DER STECKKARTE OHNE LÖTEN

rung. Beachten Sie dabei vor allen Dingen die äußerst genauen Anweisungen in der mitgelieferten zehnseitigen Bedienungsanleitung. Ebenso müssen Sie beobachten, daß Ihre Garantiebedingungen nach dem Öffnen des Computergehäuses keine Gültigkeit mehr haben (allerdings gilt das nur für Geräte, die jünger als ein halbes Jahr sind.) Ist Ihr 128 aller Abdeckplatten und sonstiger Gehäuseteile entledigt, so liegt die Platine des C128 vor Ihnen. Heben Sie nun vorsichtig den VDC-Chip heraus und bringen die Steckkarte des „Graphic Booster" entsprechend der Anweisung an, stecken Sie dann den VDC-Chip wieder an seinen alten Platz, diesmal allerdings auf der Erweiterungsplatine, zurück. Vor allen Dingen sollten Sie den Rat auf der vorletzten Seite der Bedienungsanleitung beherzigen, in der ausdrücklich darauf hingewiesen wird, die erste Blechzunge links auf dem Abschirmblech beim 128D nach oben zu biegen, um so keinen Kontakt mit dem eingesteckten Zusatzsockel zu verursachen. (Ihr 128er könnte es Ihnen sehr übel nehmen.)

Und jetzt kommt der „Hammer" (oder auch eine gute Nachricht für alle, die den „neuen" 128D besitzen): Klammheimlich hat Commodore hier nicht nur einige komfortable Veränderungen vorgenommen, nein, die Erfindung der Firma Combo ist hier bereits hardwaremäßig installiert! Alle diese grafischen Fähigkeiten stehen ab sofort zu Ihrer Verfügung, müssen allerdings mit Hilfe geeigneter Software erst geweckt werden. Interessant ist in diesem Zusammenhang, daß auch die Entwicklerfirma in der Schweiz bis vor kurzem nichts von dieser „Neuerung", die ja eigentlich ihre Erfindung ist, gewußt hat. Hat da jemand Commodore einen Tip gegeben? Egal wie, die Besitzer eines solchen 128D (er ist seit Mai 1987 im Handel und am leichtesten daran zu erkennen, daß sich der Datasettenport nicht mehr an der Gehäuserückseite, sondern seitlich links befindet), brauchen diesen etwas problematischen Hardware-Umbau nicht mehr vorzunehmen, sie erhalten die Software-Diskette von der Firma Combo um etwa 30 bis 40 Mark unter dem Neupreis des gesamten Grafikpaketes. Denn eines ist klar: ohne diese Software nützt Ihnen die neue Funktion dieses Spezial-128D auch nichts, außer,



Sie schreiben sich die entsprechende Software selber. (Doch bei der Grafikprogrammierung des VDC 8563 haben sich schon einige die Zähne ausgebissen ...)

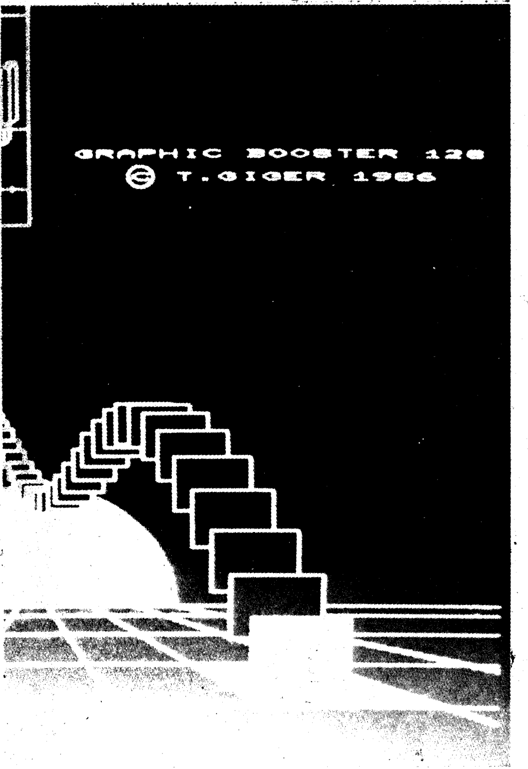
Nach dem Zusammenbau Ihres geöffneten C128 steht der „Graphik Booster" mit all seinen fantastischen Möglichkeiten zur Verfügung. Jetzt muß noch die Systemdiskette geladen werden, was mit BOOT oder per Druck auf den Reset-Taster

ERWEITERUNG PER SOFTWARE: INITIALISIEREN

geschieht. Auf dem Bildschirm sehen Sie nun die Auflösungsmodi, in die Sie Ihren 80-Zeichen-Monitor versetzen können. Die erste Zahl bedeutet die horizontale, die zweite die vertikale Auflösung. Folgende Modi stehen zu Ihrer Verfügung:

640*360 Pixel
640*720 Pixel
640*600 Pixel
720*700 Pixel
720*600 Pixel
750*600 Pixel.

Möglich macht das Ganze der Interlace-Modus, eine Verdopplung der vertikalen Auflösung. Hierbei führt der Elektronenstrahl, der die Bildschirmröhre abtastet, diesen Vorgang zweimal durch: zunächst die geraden Bildschirmzeilen, anschließend die ungeraden Linien.



Zum Vergleich: Im Bild oben die Grafikauflösung im 40-Zeichenmodus, unten die Fähigkeiten des 'Grafikboosters'.

Dabei sollten Sie allerdings beachten, daß Sie auch den entsprechenden Monitor besitzen. Der 1081 wird bei diesen Werten seine Schwierigkeiten in ständigen Bilddurchläufen äußern, die sich sehr schwer „fangen" lassen. Zumindest sollten

PROBLEME MIT DEM MONITOR?

Sie den Drehregler der vertikalen Synchronistaion im Uhrzeigersinn bis zum Anschlag drehen (beim 1081 ist er hinten angebracht), auch die Bildhöhe müssen Sie neu einstellen (eventuell mit dem Schraubenzieher neben dem Synchronisationsknopf). Läßt sich der Monitor dadurch noch immer nicht aufhalten, sollten Sie auf die höchste Auflösung verzichten und Werte nicht über 400 Punkte vertikal verwenden.

Ein entsprechender Monitor, der für den C128 entwickelt wurde, wie beispielsweise der 190, ist hierfür der idealste. Entscheiden Sie sich nun für einen Auflösungsmodus und drücken Sie die entsprechende Taste. Nach kurzer Zeit meldet sich der Computer wieder. Ab sofort stehen Ihnen die aus dem 40-Zeichen-Modus bekannten Grafik-Anweisungen zur Verfügung, wenn auch in etwas geänderter Form, Split-Screen und Multi-Color können Sie nicht einschalten, wohl aber 256 Farben von insgesamt 3000 zur gleichen Zeit darstellen. Die Anweisung „GRAPHIC 2,1" schaltet die Grafik ein (640*360, bzw. 720*360 Pixel) und löscht den Bildschirm. (Im normalen 40-Zeichen-Modus hätte er „GRAPHIC 1,1" gelautet!)

Der Color-Befehl für die Farbgebung verhält sich ebenfalls ein wenig anders, als Sie es vom Grafikmodus des VIC-Chip gewohnt sind. Der erste Parameter kann hier einen Wert von 0 bis 255, der zweite von 0 bis 15 darstellen. Für die Hintergrundfarbe ist immer die erste Zahl maßgebend, die Vordergrundfarbe wird immer in passendem Kontrast dazu angezeigt (etwa heller Hintergrund, dunkler Vordergrund oder umgekehrt). Wie beim Amiga fällt der Bildschirmrahmen völlig weg, Ihre Grafik kann sich über die gesamte Bildfläche erstrecken.

Natürlich besteht auch die Möglichkeit, so einen Grafikbildschirm auf Disk zu speichern. Hier nutzen Ihnen aber die bislang gültigen Befehle nichts.

Für ihre Stelle wurden zwei Anweisungen des BASIC 7.0 zweckentfremdet, die normal etwa ganz anderes bewirken:

SSHAPE"(Name)",8

speichert ein Bild auf Disk, je nach gewähltem Auflösungsmodus benötigt es um die 200 Blocks auf der Diskette;

GSHAPE"(Name)",8

holt das Bild wieder in den Speicher des C128 zurück und zeigt es sofort auf dem Bildschirm an.

Desgleichen wurde der CHAR-Befehl erweitert. So läßt sich jetzt ein Text in Ihrer Grafik nicht nur im Großschrift- und -Blockgrafikmodus darstellen, sondern auch in Kleinschrift, außerdem gilt hier nicht das Koordinatensystem des Textzeilen-Bildschirms (40*24), sondern das von Ihnen bei der Installation des „Graphic Boosters" gewählte. Der Clou: Der angezeigte Text kann nun mit Hilfe zweier zusätzlicher Parameter für Höhe und Breite (zwischen 0 bis 100) vergrößert werden. (Die ehemaligen C64-Freaks unter unseren Lesern werden sich sicher noch an Simon's BASIC erinnern, eine sehr komfortable BASIC-Erweiterung, bei der es auch möglich ist, Text in einer Grafik, zumindest in vertikaler Richtung, zu verändern. Dem „normalen" BASIC 7.0 des C128 fehlt leider diese Möglichkeit ...)

KOMFORTABLER „CHAR"-BEFEHL

Die beiden Soft- und Hardware-Entwickler aus der Schweiz haben noch eine recht angenehme Option dazugepackt: Scrollen um jeweils acht Zeilen in beiden Richtungen, durch einen SYS-Befehl aufrufbar. Durch seine professionelle Anwendungsmöglichkeit kann der „Graphic Booster" ohne weiteres zum Programmieren, Gestalten und Entwerfen ebenso professioneller Grafik verwendet werden.

Hier möchten wir auf alle Fälle das Arbeiten mit zwei Bildschirmen empfehlen, links Text, rechts Grafik.

Übrigens: Die bereits erwähnten Auflösungsarten sind auch unter dem Betriebssystem CP/M ansprechbar. Sind Sie glücklicher Besitzer einer Speichererweiterung (RAM-Floppy) für den C128 (1700/1750), so können Sie insgesamt acht Bilder in der höchsten Auflösung 720*700 entwerfen und bearbeiten.

FAZIT

Mit dem „Graphic Booster" stehen dem C128-Benutzer bislang ungeahnte Möglichkeiten offen, zu denen sein Computer fähig ist: neue Arten der Textdarstellung durch einen 43- oder 50-Zeilen-Editor (ohne Interlace), CAD, Desktop-Publishing und anderes mehr.

SUBROUTINES IM 128er

Dem 128er unters Hemd geschaut

Im ROM des Commodore 128 sind viele nützliche Maschinenspracheroutinen enthalten, die zu nutzen in zweifacher Hinsicht lohnt: Erstens benötigen diese Programmchen keinen Extra-Speicherplatz und zweitens sind sie erprobt, also frei von Fehlern.

BASIC ist eine in Maschinensprache programmierte Hochsprache. Die Steuerroutinen dieser Sprache liegen im ROM. Viele von Ihnen sind für Maschinenspracheprogrammierer von großem Wert. Vorausgesetzt, man weiß, welche Routinen was bewirken und wie man sie ansteuern muß. Aus diesem Grund drucken wir eine Übersicht, geordnet nach ihren Einsatzgebieten.

Abbrechen eines BASIC-Programmes ohne Break
Einsprungadresse:
$4dC6/19910
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
END

Eine der weniger sinnvollen Systemunterroutinen. Wird meist benutzt, um Verblüffung hervorzurufen. Springt ein BASIC-Programm mit SYS 19910 END an, bricht das BASIC-Programm ohne Ausdruck von „Ende" oder „Break in . . .", wie normalerweise üblich, ab.

Akkumulatorwert abspeichern
Einsprungadresse:
$FF77/65399
Benutzte Register:
Akkumulator, X, Y, Speicherstelle $2B9
Codebezeichnung:
INDSTA

Um den Akkumulatorinhalt in eine beliebige Speicherstelle zu schreiben, benutzen viele Maschinenspracheprogrammierer die INDSTA-Routine. Sie funktioniert ähnlich INDCMP, „Wertevergleich Akkumulator mit Speicherstelle". In der Nullseite des C128 steht der Pointer auf die Adresse, in die der Akkumulatorinhalt geschrieben werden soll. Adresse $2B9 zeigt auf diesen Nullseitenpointer. Das X-Register fungiert als Konfigurationsindex, das Y-Register als Offset des Nullseitenpointers.

Bildschirmausgabebreite tauschen
Einsprungadresse:
$FF5F/65375
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
SWAPPER

Gibt der C128 seine Daten auf den 40-Zeichen-Bildschirm aus, ändert Swapper die Ausgabe auf 80-Zeichen-Bildschirm um. War es vorher der 80-Zeichen-Bildschirm, der angesprochen wurde, bevor SWAPPER wirksam war, printed der 128er alle Bildschirmausgaben danach auf den 40-Zeichen-Bildschirm. SWAPPER ist nur nützlich, wenn man innerhalb eines Programmes zwei Bildschirme ansprechen will (sehr selten). Wichtig ist: Keine Verwechselungen, denn ein SWAPPER zuviel oder zuwenig bringt alle weiteren Ausgaben durcheinander.

Bootreset
Einsprungadresse:
$FF56/65366
Benutzte Register:
Keine Registersteuerung möglich.
Codebezeichnung:
PHOENIX

Um vor unbefugten Zugriffen zu schützen, oder ein Programm am Ende seiner Funktion wieder ins Hauptmenü auf Diskette springen zu lassen, nutzt man PHOENIX. Dieses Unterprogramm führt einen RESET mit anschließendem Boot-Versuch durch. Im Gegensatz zu BOOTCALL = „Diskette booten" geht die Kontrolle über den C128 entweder an das Bootprogramm oder an das Betriebssystem über. Bei BOOTCALL kontrolliert das Ursprungsprogramm weitere Aktivitäten, falls keine Boot-Filediskette eingelegt war. Aus diesem Grund ist BOOTCALL meist PHOENIX vorzuziehen, da unter Umständen keine Softwarekontrolle über den Rechner mehr möglich ist.

C64-Modus einschalten
Einsprungadresse:
$FF4D/65357
Benutzte Register:
Register werden neu gesetzt.
Codebezeichnung:
C64MODE

Um vom 128er-Modus zum C64-Modus zu gelangen, genügt der Sprung in die C64MODE-Programmroutine. Nach diesem Sprung ist eine Kontrolle über den Computer mittels Programm nicht mehr möglich, da C64MODE automatisch einen Reset durchführt.

Diskette booten
Einsprungadresse:
$FF53/65363
Benutzte Register:
X-Register
Codebezeichnung:
BOOTCALL

Möchte man ein Programm auf der Diskette booten, gibt man dem XCY-Register das zu bootende Laufwerk entsprechend dessen Geräteadresse an und springt dann in die BOOTCALL-Routine. Ist auf der eingelegten Diskette kein Bootprogramm gespeichert, bricht BOOTCALL den Boot-Versuch ab und übergibt die Kontrolle an den



Rechner zurück. Texte wie „Falsche Diskette eingelegt" können nun ausgedruckt werden, um nach Tastendruck einen erneuten Bootversuch zu starten.

Einlesen von Zeichen
Einsprungadresse:
$FFE4/65508
Benutzte Register:
Akkumulator
Codebezeichnung:
GETIN

Mit diesem Unterprogramm ist es möglich, sowohl Daten aus einer Diskettendatei, als auch Daten von der Tastatur einzulesen. Je nach Wunsch, muß vorher die entsprechende Eingabedatei (Tastatur oder Floppy oder die RS-232-Schnittstelle) angegeben werden.

IEC-Bus auslesen
Einsprungadresse:
$FFA5/65445
Benutzte Register:
Akkumulator
Codebezeichnung:
ACPTR

Nachdem diese Routine angesprungen wurde, ist bei erfolgreicher IEC-Bus-Auslese im Statusbyte der Nullseite, der Speicherstelle $90, das höchste Bit gesetzt. Der ausgelesene Wert selbst steht in diesem Fall im Prozessorakkumulator. Bei nicht gesetztem Statusbyte steht ein beliebiger Wert im Akkumulator..

IEC-Bus ansenden
Einsprungadresse:
$FFA8/65448



Benutzte Register:
Akkumulator
Codebezeichnung:
CIOUT

Entspricht als Gegenstück der IEC-Bus-Einlese-Routine. Der zu sendende Wert ist vor Ansprung dieser Subroutine in den Akkumulator zu schreiben. Nach dem Sendevorgang gibt das Statusbyte in Adresse $90 bei gesetzt an, daß der Sendevorgang geglückt ist. Ansonsten trat ein Fehler auf.

I/O-Init
Einsprungadresse:
$FF84/65412
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
IOINIT
Oftmals reagieren angeschlossene Geräte nicht im erwarteten Maße, weil bereits Unterprogramme die Datenleitungen blokkieren oder Datenflußzeiger verbogen haben. IOINIT bringt alle I/O-Leitungen in den Ursprungszustand zurück. Diese Routine bewirkt somit einen Reset aller I/O-Leitungen des 128ers.

Modi löschen
Einsprungadresse:
$C77D/51069
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
CLQIR

Um die drei besonderen Modi des C128 (Hochkomma-, Insert- und Reversmodus) garantiert zurückzusetzen, genügt es, diese Subroutine anzuspringen. Speziell bei Programmierung von Programmiersprachen und Textverarbeitungsprogrammen ist diese Unterroutine sehr wichtig.

Reset ausführen
Einsprungadresse:
$E000/57344
Benutzte Register:
Löscht Register.
Codebezeichnung:
RESET

Um alle Register, Datenleitungen und sonstige Arbeiten des C128 wirkungsvoll und auf Dauer per Softwareroutine zu unterbrechen, genügt ein kurzer Sprung in $E000. Danach führt das System einen Reset aus. Vom Ursprungsprogramm bleibt wenig übrig, die Kontrolle über den Computer geht an das Betriebssystem.

RUN-STOP-Taste abfragen
Einsprungadresse:
$FFE1/65505
Benutzte Register:
Z-Flag im Prozessor.
Codebezeichnung:
STOP

In sehr vielen Programmen kann der Anwender Funktionen mittels RUN-STOP-Taste unterbrechen, um danach ins Hauptmenü zurückzukehren. Ebenfalls in BASIC ist es möglich, den Programmablauf mit RUN-STOP zu unterbrechen, falls das System nicht hängt oder die Taste abgeschaltet ist.
Die meisten Programmierer greifen bei der Abfrage der RUN-STOP-Taste auf die STOP-Routine des Betriebssystems zurück. Nutzt man dieses Unterprogramm, so steht im Zero-Flag des Prozessors eine Eins, falls die RUN-STOP-Taste gedrückt ist. Ansonsten steht im Zero-Flag eine Null. Wurde die STOP-Taste gedrückt, macht der C128 alle OPEN-Befehle rückgängig.

Softwarereset ausführen
Einsprungadresse:
$FF87/65415
Benutzte Register:
Register werden ohne Sinn manipuliert.
Codebezeichnung:
RAMTAS

Im Gegensatz zu RESET initialisiert RAMTAS nicht alle Funktionen des Computers, sondern setzt nur Zero-Page-Pointer auf Anschaltanfangswerte zurück und löscht die Kassetten- und Tastaturpuffer. Um den C128 vollständig zu initialisieren, genügt $E000/57344.

Sprung in beliebige Speicherblöcke
Einsprungadresse:
$FF71/65393
Benutzte Register:
Register bleiben erhalten.
Codebezeichnung:
IMPFAR

Durch die Bank-Speichereinteilung ist der Speicheraufbau des C128 komplizierter, als der des C64. IMPFAR umgeht einige der dadurch auftretenden Schwierigkeiten. Gemäß den, in den Nullseitenadressen $02 bis $09 angegebenen Werten, führt der Computer einen Sprung an eine andere Speicherstelle durch, um dort ein Unterprogramm zu bearbeiten. Die *Zieladresse* selbst ist wie folgt festgelegt:

Konfigurationsindex steht in: $02
Programmzähler ist enthalten in: $03/04
Prozessorstatus steht in: $05
Akkumulator, X- und Y-Register: $06/07/08
SP=Stackpointer: $09

Die Routine selbst gehört allerdings nicht zu den schnellsten.

Sprung in Subroutine mit beliebigem Anfang
Einsprungadresse:
$FF6E/65390
Benutzte Register:
Register bleiben erhalten.
Codebezeichnung:
JSRFAR

Um Unterprogramme in anderen Speicherblöcken anzuspringen, ohne dabei programmiertechnische Klimmzüge vollführen zu müssen, nutzt man diese Routine. Sie ist mit JMPFAR=Sprung in beliebige Speicherblöcke, zu vergleichen. Das *Sprungziel* ist hier ebenfalls durch die Nullseitenadressen $02 bis $09 definiert:

Konfigurationsindex steht in: $02
Programmzähler ist enthalten in: $02/$03
Prozessorstatus steht in: $05
Akkumulator, X- und Y-Register: $06/07/08
SP=Stackpointer: $09

Statusbyte auslesen
Einsprungadresse:
$FFB7/65463
Benutzte Register:
Akkumulator
Codebezeichnung:
READST

Das Statusbyte in Adresse $90 ist eine oft benötigte Hilfe bei der Kontrolle einiger Unterroutinen. Speziell Unterprogramme mit IEC-Bus-Schreib- oder Lese-Befehlen nutzen $90, um über den Erfolg der Operation zu berichten. (Bei Erfolg ist das Statusbyte gesetzt.

READST hilft dem Programmierer, indem nach Ansprung dieser Unterroutine der Statusbytewert im Akkumulator steht und dort für weitere Vergleiche genutzt werden kann.

Systemnachrichten manipulieren
Einsprungadresse:
$FF909/65424
Benutzte Register:
Akkumulator
Codebezeichnung:
SETMSG

Der Commodore 128 hat zwei Möglichkeiten im Umgang mit Systemmeldungen: 1. Er printed sie unter allen Umständen aus; 2. Er unterdrückt sie in jedem Fall. In bestimmten Fällen ist es sinnvoll, die Ausgabe der Systemmeldungen derart zu manipulieren.
Das Kontrollbyte über diese Funktion steht in $9D der Zero-Page. Ist das höchste Bit dieser Adresse gesetzt, werden Systemmeldungen ausgegeben. Ist das höchste Bit ungesetzt, unterdrückt der 128er sämtliche Systemmeldungen. Mit SETMSG schreibt man den Wert des Akkumulators direkt nach $9D.
Ist der Akkumulatorwert größer 128, sind alle Systemmeldungen nach SETMSG erlaubt. Ist der Akkumulatorwert geringer als 128, unterdrückt der Commodore nach SETMSG weitere Systemmeldungen.

Systemuhr auslesen
Einsprungadresse:
$FFE1/65502
Benutzte Register:
Akkumulator, X-, Y-Register.
Codebezeichnung:
RDTIM

Im Commodore 128 steht eine Systemuhr zur Verfügung. Viele Programmierer greifen auf diese (relativ ungenaue aber einfach zu handhabende) Zeitmessungsmethode zurück.
Insbesondere Grafik- und Textadventures benötigen weniger exakte Zeitmessung und nutzen daher verstärkt RDTIM.
Nachdem diese Routine durchlaufen wurde, stehen die Zeitwerte in:
Akkumulator = Stundenwert, X-Register = Minutenwert, Y-Register = Sekundenwert.
Um die Uhr zu setzen, nutzt man SETTIM in Adresse $FFDB/65499.
Die abzuspeichernden Werte sind in gleicher Art und Weise auf Akkumulator, X- und Y-Register verteilt.

Systemuhr setzen
Einsprungadresse:
$FFDB/65499
Benutzte Register:
Akkumulator, X-, Y-Register.
Codebezeichnung:
SETTIM

Um die, im C128 vorhandene, automatische Systemuhr zu setzen, bietet sich diese Routine an.
Der zu setzende Stundenwert ist vorher in den Akkumulator, der Minutenwert in das X-Register und der Sekundenwert in das Y-Register zu schreiben.
Nach Ansprung der Unterroutine sind diese Parameter übergeben und der 128er beginnt automatisch, ab diesem Zeitpunkt weiterzuticken.
Mit dem Unterprogramm „Systemuhr auslesen"= RDTIM liest der C128 die Systemuhr aus und überträgt die Daten entsprechend der SETTIM-Routine in Akkumulator, X- und Y-Register.

Tastatureingabe prüfen
Einsprungadresse:
$FF9F/65439
Benutzte Register:
Registerwerte bleiben gleich.
Codebezeichnung:
KEY

Um zu testen, ob eine Taste gedrückt ist, springt man die KEY-Unterroutine an. Im Falle eines Tastendruckes wird der Tastenwert in den Tastaturpuffer geschrieben und die Pufferspeicherzahl um den Wert 1 erhöht. Erwähnenswert ist: KEY testet automatisch, in welchem Zustand (ob in ASCII oder DIN) sich die Tastatur augenblicklich befindet und nimmt den Wert aus der entsprechenden Tabelle.

Tastaturwerte bis CR einlesen
Einsprungadresse:
$C258/49752
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
GETLIN

Jedes über Tastatur eingegebene Zeichen wird angenommen und am Bildschirm geprinted, bis Return gedrückt wurde. Wichtige Routinen für Texteingaben.

Textblöcke ausgeben
Einsprungadresse:
$FF7D/65045
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
PRIMM

Eine der wichtigsten Routinen, um Texte auszugeben, ist PRIMM. Das Prinzip von PRIMM ist sehr einfach: Man schreibt in den Speicher den JSR $FF7D und danach einen Text beliebiger Länge mit beliebigen Steuerzeichen wie Cursor up und Return. Der C128 printed diesen Text auf das jeweilige aktuelle Ausgabegerät wie Bildschirm, Drucker oder Laufwerk. PRIMM hört erst dann auf, Daten an das Ausgabegerät zu senden, wenn es auf ein Byte mit dem Wert Null stößt. PRIMM ist damit eines der mächtigsten Ausgabewerkzeuge eines Maschinenspracheprogrammierers.

Wertevergleich Akkumulator mit Speicherstelle
Einsprungadresse:
$FF7A/65402
Benutzte Register:
Akkumulator, X, Y, Speicherstellen $2C8, $05.
Codebezeichnung:
INDCMP

Mit dieser Routine vergleicht man den Wert des Akkumulators mit dem Wert einer beliebigen Speicherstelle. Diese Speicherstelle wird wie folgt angegeben: In der Speicherstelle $02C8 steht der Pointer auf der Adresse, die den zu vergleichenden Wert enthält. Das X-Register fungiert als Konfigurationsindex, das Y-Register als Offset des Nullseitenpointers. In Speicherstelle $05 ist das Vergleichsergebnis nach der Operation zu finden.

Zeichenmodus setzen
Einsprungadresse:
$FF81/65409
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
CINIT

CINIT ist ein mächtiges Unterprogramm mit entsprechend vielen Funktionen: 1. CINIT prüft nach, ob gerade der 40- oder der 80-Zeichen-Modus per Tastatur eingestellt ist und initialisiert die Ausgabe auf den entsprechenden Wert. Neben der Änderung des Videokontrollers werden alle Funktionstastenbelegungen gelöscht oder neu initialisiert.
Daten im Tastaturpuffer und der dazugehörige Pointer werden auf Null gesetzt und alle I/O-Kanäle automatisch geschlossen. Maschinensprache-Programmierer müssen aufpassen, daß sie bei gewünschten Funktionen dieser Routine die ungewünschten Neben- und Folgeeffekte nicht vergessen, sondern sich dagegen absichern.

Character-ROM ins VDC-RAM kopieren
Einsprungadresse:
$FF62/65360
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
DLCHR

DLCHR kopiert, je nach DIN-ASCII-Einstellung,

den jeweiligen Zeichensatz vom Charakter-ROM ins VDC-RAM. Im RAM läßt sich der Zeichensatz beliebig verändern und nutzen. Notwendig hierfür ist jedoch, die Ausgabe auf 80 Zeichen je Zeile zu stellen.

CHKIN und CKOUT zurücksetzen
Einsprungadresse:
$FFCC/65484
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
CLRCH
Wurden mit CHKIN und CKOUT Kanalnummern verändert und möchte man diese Änderungen rückgängig machen, springt man CLRCH an. CLRCH setzt alle Änderungen zurück und sendet zugleich einen Talk und einen UNTALK-Befehl, um alle vormals angesprochenen Geräte wieder in deren Wartezustand zurückzuversetzen.

Cursor-Home
Einsprungadresse:
$C150/49752
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
CURHOM

Die meisten BASIC-Programmierer nutzen von BASIC aus diese Subroutine regelmäßig. Für Maschinensprache-Programmierer ist sie unumgänglich.
Ist vor Nutzung dieser Routine ein Fenster definiert, führt CURHOM den Cursor an die linke, obere Ecke des definierten Fensterbereiches. Ansonsten entspricht CURHOM dem Tastendruck der CLR-HOME-Taste.

Cursorstandpunkt manipulieren/feststellen
Einsprungadresse:
$FFF0/65520
Benutzte Register:
X- und Y-Register, Carry-Flag.
Codebezeichnung:
PLOT

Auch diese Unterroutine arbeitet je nach Carry-Flag in zwei verschiedene Richtungen: Ist das Carry-Flag gesetzt, während PLOT aufgerufen wurde, so steht nach Ablauf der Prozedur im X-Register der Wert der aktuellen Cursorspalte und im Y-Register der Wert der aktuellen Cursorzeile. Ist das Carry-Flag gelöscht, dann setzt PLOT den Cursor an die Stelle: Zeile = Y-Register-Wert und Spalte = X-Register-Wert. PLOT zu nutzen lohnt sich vor allem vor positionierten Ausgaben mit PRIMM, BSCOUT oder BSOUT SCRN in den Adressen $FF7D/65405, $FFD2/65490 und $C72D/50989, die zur Ausgabe von Texten ab der aktuellen Cursorposition dienen.

Dateien schließen
Einsprungadresse:
$FFE7/65511
Benutzte Register:
Register bleiben gleich.
Codebezeichnung;
CLALL

CLALL schließt alle gerade geöffneten Dateien und Kanäle. Danach springt es selbständig eine andere Unterroutine an. Diese Unterroutine ändert die Kanaleinlese- und -ausgabenummern auf den Anfangszustand nach dem Einschalten um.

DMA-Kontroller nutzen
Einsprungadresse:
$FF50/65360
Benutzte Register:
X-Register.
Codebezeichnung:
DMACALL

Um DMA-Zugriffe zu steuern, setzt man DMACALL ein. Im X-Register steht der hierfür neue Konfigurationswert. Im Normalfall ist diese Routine unwichtig, da meist kein Gerät am C128 angeschlossen ist, das diese Routine nutzen kann. Beispielsweise wäre ein solches Gerät eine Festplatte (=Harddisk).

Eingabe-/Ausgabebereichs-Adressen einlesen
Einsprungadresse:
$FFF3/65523
Benutzte Register:
X- und Y-Register.
Codebezeichnung:
IOBASE

Nach Anspringen dieses Unterprogramms befinden sich im X-Register das Low-Byte und im Y-Register das High-Byte der Basis-Eingabe-Ausgabe-Bausteine-Adresse ($D000).

Empfang abschalten
Einsprungadresse:
$FFAE/65454
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
UNLSN
Um externe Geräte auf Empfang zu schalten, benutzte man die LISTEN-Routine. Mit UNLSN unterbricht man die Empfangsbereitschaft des Gerätes.

Empfang initialisieren
Einsprungadresse:
$FFB1/65457
Benutzte Register:
Akkumulator.
Codebezeichnung:
LISTEN

Um ein externes Gerät auf Empfang zu stellen, springt man die Subroutine LISTEN an. Das anzusprechende Gerät wird durch den Wert im Akkumulator (Geräteadresse) bestimmt. Danach ist das Gerät solange auf Empfang gestellt, bis die Routine UNLSN angesprungen wird. Um Daten an das Gerät auszugeben, ist die CIOUT-Routine in $FFA8/65448 zu verwenden.

Fensterbereich löschen
Einsprungadresse:
$C142/49474
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
CLRWIN

Um den Textinhalt in einem definierten Fenster zu löschen, nutzt man die CLRWIN-Routine. Im Falle keines definierten Textfensters nimmt CLRWIN den gesamten Bildschirm als Fenster an und löscht diesen. Allerdings benötigt diese Routine mehr Zeit als die üblicherweise benutzte CLR-HOME-Routine.

Funktionstasten umbelegen
Einsprungadresse:
$FF65/65381
Benutzte Register:
Akkumulator, X- und Y-Register.
Codebezeichnung:
PFKEY

Der Commodore 128 bietet die Möglichkeit, seine zehn Funktionstasten und die HELP-Taste mit anderen Funktionen oder Texten frei zu belegen. Dafür gibt es teilweise Programme, teilweise ist es in Programmen wie Textverarbeitungen schon eingebaut. Sinnvoll ist es allemal und als Maschinensprache-Programmierer nutzt man in diesem Fall die PFKEY-Routine.
Die Handhabung ist folgendermaßen: Im X-Register steht die Nummer der zu belegenden Funktionstaste, also 1 bis 12 oder $01 bis $0C. Im Y-Register steht die Länge der Zeichenkette, die der Funktionstaste zugeordnet werden soll.
Der Text selbst wird durch ein Nullseitenpointer angegeben, dessen Speicherstelle wiederum im Akkumulator steht.

Kanal schließen
Einsprungadresse:
$FFC3/65475
Benutzte Register:
Akkumulator, Carry-Flag.
Codebezeichnung:
CLOSE

CLOSE schließt eröffnete Kanäle ab. Die gewünschte Kanalnummer, die zu schließen gilt, muß vorher in den Akkumulator geschrieben werden. Falls die Routine fehlerhaft arbeitet, setzt sie das Carry-Flag des Mikropro-

zessors. Damit läßt sich beispielsweise prüfen, ob ein Kanal überhaupt geöffnet ist.

Kanalausgabenummer festlegen
Einsprungadresse:
$FFC9/65481
Benutzte Register:
X-Register und Carry-Flag.
Codebezeichnung:
CKOUT

Das logische Gegenstück zu CHKIN. Um mit BSOUT und dem Akkumulatorwert Daten auszugeben, muß man vorher den gewünschten Ausgabekanal mittels der CKOUT-Routine bestimmen. Das anzuwählende Gerät steht als Gerätenummer = Kanalnummer im X-Register.
Falls kein Ausgabegerät der angesprochenen Nummer gefunden wurde, setzt CKOUT das Carry-Flag des Prozessors (beispielsweise bei abgeschaltetem Drucker und dem Versuch, dieses Gerät anzusprechen).
Um unnötige Programmabbrüche zu vermeiden, ist es sinnvoll, das Carry-Flag nach Nutzung der CKOUT-Routine zu prüfen.

Kanal-Einlesenummer festlegen
Einsprungadresse:
$FFC6/65409
Benutzte Register:
X-Register und Carry-Flag.
Codebezeichnung:
CHKIN

Um die Routine BASIN ($FFCF/65487) richtig zu nutzen, muß man vorher mittels CHKIN die Kanal-Einlesenummer festlegen. Die gewünschte Kanalnummer, also das gewünschte Einlesegerät, ist mit dem X-Register festzulegen. Konnte der Kanal nicht eröffnet werden, setzt CHKIN das Carry-Flag des Prozessors. Ist das Carry-Flag = 0, so wurde das gewünschte Einlesegerät gefunden und angesprochen.

Konfigurationstabelle auslesen
Einsprungadresse:
$FF6B/65387
Benutzte Register:
X-Register.
Codebezeichnung:
GETCONF

Um die 128 KByte des C128-Speichers zu verwalten, nutzen viele Anwender die GETCONF-Unterroutine. GETCONF speichert im Akkumulator das, dem X-Register entsprechende, Konfigurationsbyte aus der Tabelle ab $F7F0.
Vorsicht: Steht im X-Register ein höherer Wert als 16 (maximale Wertezahl), kann es zu unvorhersehbaren Systemfehlern kommen.

Senden/initialisieren
Einsprungadresse:
$FFB4/65460
Benutzte Register:
Akkumulator.
Codebezeichnung:
TALK
Um ein externes Gerät, das am IEC-Bus angeschlossen ist, zum Senden seiner Informationen zu bringen, springt man die Unterroutine TALK an. Die Geräteadresse des gewünschten Sendegerätes muß vorher im Akkumulator angegeben werden. Um die Sendequelle wieder auf Wartestellung zu bringen, springt man die UNTLK-Unterroutine in $FF96/65430 an.

Sender abschalten
Einsprungadresse:
$FFAB/65451
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
UNTLK

Wurde ein Gerät mit dem Unterprogramm TALK zum Senden von Daten angeregt, schaltet man die Sendungen mit UNTLK wieder ab. Während in TALK im Akkumulator die Gerätenummer anzugeben ist, sendet UNTLK ein generelles Sendestop.

Speicherobergrenze abfragen/bestimmen
Einsprungadresse:
$FF99/65433
Benutzte Register:
X- und Y-Register, Carry-Flag.
Codebezeichnung:
MEMTOP

Mit MEMTOP liest man entweder aus den Nullseiten-Pointeradressen die Werte der oberen Speichergrenze oder schreibt die gewünschten Werte dieser Grenze in diese Speicherstellen: Ist das Carry-Flag gesetzt (SEC), dann nimmt die Speicherobergrenze den Wert von X-Register = Low-Byte, und Y-Register = High-Byte an.
Es ist erforderlich, vorher zu prüfen, ob die Speicheruntergrenze weit genug unter dem Wert der neuen Speicherobergrenze liegt, da es ansonsten zu Systemabstürzen kommen kann.
Ist das Carry-Flag des Prozessors gelöscht (CLC), dann liest MEMTOP die Speicherobergrenze in: High-Byte = Y-Register, Low-Byte = X-Register. Die Untergrenze des Speichers erfährt man durch MEMBOT in $FF9C/65436.

Speicheruntergrenze abfragen/bestimmen
Einsprungadresse:
$FF9C/65436
Benutzte Register:
X- und Y-Register, Carry-Flag.
Codebezeichnung:
MEMBOT

Mit MEMBOT liest man entweder aus den Nullseiten-Pointeradressen die Werte der unteren Speicherbegrenzung oder man schreibt die gewünschten Untergrenzwerte in diese Speicherstellen: Ist das Carry-Flag gesetzt (Maschinensprachebefehl: SEC), dann nimmt die Speicheruntergrenze den Wert von X-Register= Low-Byte und Y-Register =High-Byte an.
Es ist unbedingt erforderlich, vorher zu prüfen, ob die Speicherobergrenze weit genug über dem Wert der neuen Speicheruntergrenze liegt, da es ansonsten zu Systemabstürzen kommen kann.
Ist das Carry-Flag des Prozessors gelöscht (Maschinensprachebefehl: CLC), dann liest MEMBOT die Speicheruntergrenze in: High-Byte=Y-Register, Low-Byte=X-Register.
Die Untergrenze des Speichers erfährt man durch die Unterroutine MEMTOP in $FF99/65433.

Speicherzelle lesen
Einsprungadresse:
$FF74/65396
Benutzte Register:
Akkumulator, X- und Y-Register.
Codebezeichnung:
INDFET

Der Speicher des Commodore 128 ist in mehrere Speicherblöcke aufgeteilt. Üblicherweise kann man eine Speicherstelle einer anderen Bank nur dadurch ansprechen, daß man zwischen der gerade aktuellen Bank zur gewünschten Bank hinschaltet, um den Wert zu lesen und dann wieder in die alte Bank zurückschaltet.
INDFET löst dieses Problem auf einfache Weise: Im X-Register ist zuvor der Konfigurationsindex und im Y-Register der Nullseitenpointer-Offset einzupoken.
Im Akkumulator steht der Wert der Nullseitenspeicher-Adresse, die auf den zu lesenden Speicher zeigt. Der gelesene Wert steht am Ende der Prozedur im Akkumulator.
Die Adresse errechnet sich aus: Pointeradresse + Wert des Y-Registers in der Speicherbank, bestimmt durch das X-Register.

Systemuhr hochzählen
Einsprungadresse:
$FFEA/65514
Benutzte Register:
Register bleiben gleich.
Codebezeichnung:
UDTIM

Im Commodore 128 befindet sich eine interrupt-gesteuerte Uhr. Dieses Zeitmeßinstrument ist nicht genau, da bei Zugriffen auf die Diskette beispielsweise die Interruptzyklen verändert werden und die Uhr daher in falschen Zeitabständen hochzählt.
Um diese Fehlerquote etwas herabzusetzen, kann man die IRQ-gesteuerte Uhr inkrementieren und somit durch Diskettenzugriffen hervorgerufene falsche Zeiten der richtigen anpassen.

Systemvektoren initialisieren
Einsprungadresse:
$FF8A/65418
Benutzte Register:
Carry-Flag.
Codebezeichnung:
RESTOR

Gerade in der ersten Testphase eines Assemblerfreaks wird der Computer gnadenlos zusammengepoked. Registerwerte werden verändert, Ein- und Ausgabekanäle verbogen; selbst die Systemvektoren sind in diesen Fällen nicht heilig. Allerdings kann es vorkommen, daß eine unbedachte Systemvektormanipulation den Rechner nahe an den Rand des Absturzes, wenn nicht noch weiter, bringt.
Um total vermanipulierte Systemvektoradressen mittels eines Befehls oder Sprungs in ein Unterprogramm wieder funktionsfähig zu machen, nutzt man RESTOR. Nach RESTOR entsprechen alle Systemvektoren wieder den Daten nach Anschalten des Computers. Nach RESTOR ist das Carry-Flag des Prozessors gelöscht.

Zeichen auf Bildschirm ausgeben
Einsprungadresse:
$C72D/50989
Benutzte Register:
Akkumulator.
Codebezeichnung:
BSOUT SCRN

Fast jedes Maschinenprogramm muß interaktiv Daten auf den Bildschirm schreiben und Daten von der Tastatur einlesen. BSOUT SCRN ist die schnellste und einfachste Methode, um einzelne Zeichen auf den Bildschirm zu leiten. Das jeweilige Zeichen muß als Code-Wert im Akkumulator stehen. Die Ausgabe selbst ist am aktuellen Cursorstandpunkt.

Zeichen aus CHKIN-definierter Datei lesen
Einsprungadresse:
$FFCF/65487
Benutzte Register:
Akkumulator.
Codebezeichnung:
BASIN
Bevor man mit BASIN kontrollierte Zeichen aus einer Datei oder der Tastatur oder dem Bildschirm einlesen kann, muß man die Lesequelle mit „Kanaleinlesenummer festlegen" = CHKIN, bestimmen. Danach liest BASIN bei jedem Sprung das aktuelle Zeichen der Lesequelle ein. le ein.

Zeichen ausgeben
Einsprungadresse:
$FFD2/65490
Benutzte Register:
Akkumulator.
Codebezeichnung:
BSOUT
BSOUT ist das logische Gegenstück zu „Zeichen aus CHKIN definierter Datei lesen" = BASIN. Das auszugebende Zeichen muß vorher in den Akkumulator gelesen werden.
Ist als Ausgabekanal der Bildschirm definiert und möchte man den CBM-Code sehen, wird der ASCII-Akkumulatorwert in den entsprechenden CBM-Code-Wert umgerechnet. Das Ausgabegerät selbst ist zuvor mit CKOUT = „Kanalausgabenummer festlegen" zu bestimmen.

Fenstergröße lesen
Einsprungadresse:
$C00F/49167
Benutzte Register:
Akkumulator, X- und Y-Register.
Codebezeichnung:
SCROG

Der Commodore 128 stellt die Möglichkeit zur Verfügung, Textfenster zu definieren. Um zu wissen, welche Ausmaße das gerade aktuelle Fenster besitzt, springt man SCORG an. Nach dem Ansprung findet man im X-Register die maximale Anzahl der zulässigen Spalten, im Y-Register in maximale Anzahl der Zeilen und im Akkumulator die gerade eingestellte Ausgabeart (40- oder 80-Zeichen-Ausgabe auf den Bildschirm).

Sven Faulhaber





Durben/Plenge/Löffelmann/Vüllers:
Das große Grafikbuch zum C128.
Düsseldorf: Verlag Data Becker.
400 Seiten.
ISBN 3-89011-154-8.
39 Mark.

Nicht zuletzt (oder gerade wegen) seiner vielgepriesenen Grafikeigenschaften haben sich die Autoren in diesem Buch recht ausgiebig mit dem C128 beschäftigt. Derjenige, der mit dem Handbuch gewappnet, sich die Grafik-Wunderwelt seines 128ers erschließen will, sollte dies schnell beiseite legen und sich in das Grafik-Buch vertiefen.

Zunächst werden sämtliche Grafikbefehle des BASIC 7.0 ausführlich anhand von Beispielprogrammen erklärt. Nun weiß auch der Einsteiger schon eine Menge mehr, als ihm das Handbuch je hätte vermitteln können. Hat er sich dann auch noch durch das recht umfangreiche Kapitel über den VIC-II-Chip für den 40-Zeichen-Bildschirm hindurchgearbeitet, fällt es auch ihm nicht mehr schwer, Bitmaps, Hires-Grafiken und Sprites zu erstellen.

ALLES ÜBER DIE BEIDEN VIDEO-CHIPS

Zur Definition eines individuellen Zeichensatzes können Sie einen komfortablen Editor, genannt Char-Designer, verwenden (Sie müssen halt das Listing im Buch abtippen). Keine Angst: Auch die Anhänger des 80-Zeichen-Bildmodus, der durch den anderen Videochip VDC 8563 gesteuert wird, kommen in keiner Weise zu kurz.

Neben einer ausführlichen Beschreibung der einzelnen Register (es sind genau 37) finden Sie auch hier einen Charakter-Generator, der eine Zeichensatzänderung (etwa deutsche Umlaute, Sonderzeichen oder ähnliches) fast zum Kinderspiel werden läßt.
Allerdings: Ein wenig Eintipparbeit muß auch hier sein (es sei denn, Sie bestellen sich die Diskette zum Buch).

VIELE LISTINGS ZUM ABTIPPEN

Selbstverständlich fehlt es auch nicht an einer Beschreibung, wie der 80-Zeichen-Screen in hochauflösende Grafik umgeschaltet werden kann.
Aber genug der Grafik-Spielereien, jetzt wird's ernst: Ab dem nächsten Kapitel wird vor allen Dingen auf die praktische Anwendung solcher Grafikmöglichkeiten eingegangen (wie Statistikbalken, Torten- und Säulengrafik, Umsatzdiagramme).
Interessante Beispielprogramme, die natürlich jeder nach seinem Gutdünken modifizieren kann, unterstreichen anschaulich die vielfältigen Anwendungsarten. Sehr gut gelungen ist dem Autoren die Einführung in CAD mit dem Programm CAD System 128, einem Mal- und Zeichenprogramm zum Abtippen.
Zu guter Letzt wird auch noch an die Assembler-Programmierer gedacht. Der Abschnitt über Extended Graphic 128 (ebenfalls mit Listing zum Abtippen), verdeutlicht recht eindrucksvoll, daß in Assembler geschriebene Programme eben doch immens schneller sind als dieselben Befehle im BASIC 7.0
Fazit: Wer grafisch mehr aus seinem C128 machen möchte, sollte dieses Buch seiner Computerbibliothek einverleiben. Es ist in jedem guten Computershop erhältlich. *hb* □

GRAFIK-PROGRAMMIERUNG C128

Heimo Ponnath:
Grafikprogrammierung C128.
Haar: Verlag Markt & Technik.
ISBN 3-89090-202-2.
52 Mark inkl. Diskette.

Die faszinierende Welt der Grafik zu erforschen ist das Anliegen dieses Buches. Kein Home-Computer der Acht-Bit-Klasse von Commodore eignet sich besser dazu als der C128.
Mit dem BASIC 7.0 dieses Computers stehen dem Benutzer siebenundzwanzig Anweisungen zur Verfügung, die sich speziell mit den grafischen Möglichkeiten im 40-Zeichen-Modus befassen.
Der wichtigste wird auch sofort im ersten Kapitel behandelt: GRAPHIC. Was die einzelnen Modi bewirken, wie Sie die Hires- oder Multicolorgrafik einschalten und wie sich sogenannte Splitscreens (Bildschirm, in Grafik- und Textdarstellung geteilt) erstellen lassen.
Nicht minder wichtig für die Darstellung ansprechender Grafikbilder sind die Farbanweisungen mit COLOR.

GEHEIMNISSE DES FARBSPEICHERS

Umfassend wird der Farbspeicherbereich des Computers beschrieben, welche entsprechenden Byte in ganz spezifischen Speicherstellen einzutragen sind, garniert mit kleinen Beispielprogrammen.
Keine Angst, Sie brauchen diese nicht abzutippen, sie befinden sich alle auf der Demo-Diskette, die dem Buch beiliegt.
Anschauliche Grafiken verdeutlichen die Funktionsweise der Zeichenbefehle wie DRAW, BOX, CIRCLE und PAINT. Wer bisher mit dem Befehl SCALE nach den Erläuterungen im Handbuch gewisse Schwierigkeiten hatte, sollte die entsprechenden Seiten darüber in diesem Buch lesen, dann dürfte alles klar sein.
Ein recht umfangreiches Kapitel widmet sich den beweglichen Objekten, die Sie mit dem C128 darstellen können: Sprites und Shapes.
Nachdem Sie sich da durchgearbeitet haben, sollte es für Sie nur noch ein Kinderspiel sein, solche Kobolde über den Bildschirm flitzen zu lassen.

JETZT WIRD'S LEBENDIG: SPRITES

Damit der „intellektuelle Touch" nicht zu kurz kommt (das ist gewiß nicht ironisch gemeint), behandelt der Autor sehr ausführlich das Wesen der sogenannten Fractals und Apfelmännchen, Grafiken im Multicolor-Modus, die mit einem sehr komplizierten Rechenprinzip arbeiten und zu ihrer Fertigstellung oft zehn Stunden und länger brauchen.
Allerdings, wenn sie danach als fertiges Bild abgespeichert sind, können Sie in Nullkommanichts geladen werden und sich in ihrer ganzen Farbenvielfalt dem Betrachter präsentieren.
Die Väter der Grafikdarstellung mit dem C128, die Videochips VIC-II (40-Zeichen-Modus) und VDC (80-Zeichen-Darstellung) werden mit ihren wichtigsten Eigenschaften vorgestellt, bevor sich das Buch mehr allgemeinen Fragen zum C128 widmet (Speicheraufteilung, Tastatur).
Untermalt wird dieser Teil von vielen nützlichen Tips und Tricks, die für den Programmierer recht hilfreich sein können (Tastatur-Direktmodus, OLD- und MERGE-Routine, Zeichensatzänderung).
Die abgedruckten Beispielprogramme dienen dem Leser lediglich dazu, den Programmablauf nachzuvollziehen, jedes noch so kleine Listing ist auf der Diskette zum Buch abgespeichert, eine Tatsache, die schon lobend hervorgehoben werden sollte.

HILFREICHE TIPS UND TRICKS

Wer mit dem Handbuch des C128 auf Kriegsfuß steht und trotzdem gerne Grafik programmieren möchte, dem kann dieses auch für den Anfänger verständliche Buch wärmstens empfohlen werden.
Ein Wort zum Autor: Heimo Ponnath, Physikochemiker und freier Journalist, weiß, wovon er spricht. Sein bereits einige Jahre früher erschienenes Werk „Wunderland der Grafik im C64" hat unter den Freaks dieses Computers viele Freunde gefunden. *hb* □

STARPAINTER 128 – ERSTES MAL- UND ZEICHENPROGRAMM FÜR DEN 128

Für den Commodore 64 gibt's erstaunlich viele Grafiktools, mit denen man Bilder selbst per Joysticks entwerfen, sie speichern oder zur Weiterverarbeitung (oder auch nur zum Ansehen) wieder in den Computer laden kann. In den meisten Fällen lassen sich diese Grafiken auch noch schwarz auf weiß mit einem passenden Drucker zu Papier bringen. Leider haben alle einen kleinen Nachteil: sie laufen eben nur im C-64-Modus. Das gilt (oder besser gesagt: galt) auch für einen Hit unter diesen Programmen – Starpainter aus dem Sybex-Verlag, Düsseldorf.

Nun, die Verkaufszahlen des Commodore 128 haben die Software-Autoren Hansen und Sonnenschein sicher darüber nachdenken lassen, ob sich solch ein Grafikprogramm nicht auch für den C 128 verwirklichen ließe, und nun ist es fertig: Der Starpainter 128. (Übrigens bis jetzt das einzige professionelle Mal- und Zeichenprogramm für den großen Bruder des C 64.)

AUSGEZEICHNETES HANDBUCH – DISKETTEN-KOPIERPROGRAMM GRATIS

Zum Software-Paket wird auch ein gut dokumentiertes Handbuch mitgeliefert, das sich sehr wohltuend von anderen Erzeugnissen dieser Art abhebt. Es besteht nämlich nicht nur aus lauter Ausdrücken, die sowieso nur Freaks verstehen, sondern wendet sich auch speziell an die Anfänger und Computer-Neulinge. Wie man das Directory einer Diskette lädt, sollte man allerdings schon wissen ... Zum Arbeiten mit Starpainter 128 muß unbedingt eine Kopie als Arbeitsdiskette angelegt werden. Dazu dient ein relativ schnelles Kopierprogramm (nur drei Diskettenwechsel), das natürlich auch zum Vervielfältigen anderer Disks verwendet werden kann. (Man kann es auch einzeln über den Sybex-Verlag beziehen!) Die Originaldisk des Malprogramms hat einen Schreibschutz, so daß sich schwerlich die vorher unbedingt notwendigen Installationen z. B. für den Drucker vornehmen ließen, ganz zu schweigen vom Abspeichern von Bildern oder selbstentworfenen Zeichensätzen ... Eine Macke hat das Software- Paket auf Disk allerdings, die wir nicht unerwähnt lassen wollen: Beim „Booten"

PROBLEME MIT DER FLOPPY 1541

des Hauptprogrammes von Disk unter Benutzung einer Floppy 1541 landeten wir ständig beim Programmzählerstand (PC) 05 im eingebauten Maschinensprache-Monitor des C 128, wohingegen bei Verwendung einer Floppy 1571 alles wunderbar klappte. Dabei war es völlig egal, ob wir nun die Originaldisk oder die (wie oben besprochene) Arbeitskopie verwendeten. Vermutlich haben die Programmautoren hier das DOS der 1571 mit in den Ladevorgang einbezogen, ohne zu bedenken, daß es noch eine ganze Menge C-128-Besitzer gibt, die ihre gute alte Floppy 1541 nicht gleich beim Kauf des neuen Computers auf den Müll geschmissen haben (was unserer Meinung nach auch gar nicht nötig ist!). Ins Hauptprogramm kommt man zwar trotzdem, indem man vorher das File „Installation" lädt und die dortigen Fragen zu Zeichenfarbe, Eingabegerät und Druckerparameter beantwortet, aber das Ganze immer wieder von Neuem vorher durchzuexerzieren, ist dann doch zu mühselig, vor allen Dingen, weil das File „Installation" nach der Einstellung durch den Benutzer wieder so auf Disk zurückgeschrieben wird und künftig für den Betrieb von Starpainter gilt. Hier sollten sich die Programmautoren nochmals dran setzen und dies benutzerfreundlich gestalten, denn sicher wissen auch sie, daß es noch eine ganze Menge C-128-Fans gibt, die nur die Floppy 1541 besitzen. Doch genug der harten Worte, es gibt auch (natürlich!) eine ganze Menge erfreulicher Sachen zu berichten.

KEIN STUPIDES ZEICHENPROGRAMM

Starpainter 128 ist kein ausschließliches Zeichenprogramm (und will es auch gar nicht sein!), sondern eher eine sehr komfortable Grafikhilfe mit anschließender Möglichkeit zur Ausgabe auf Drucker, wie es auch der Sinn z.B. von Textverarbeitungsprogrammen ist. Dies beweist eine lange Liste von Druckern, die mit Starpainter 128 eingesetzt werden können. Man muß schon ein sehr exotisches Gerät besitzen,

NAHEZU ALLE DRUCKER KOMPATIBEL

um mit dem Bildausdruck Pech zu haben. (In unserer Redaktion klappte es mit einem Mannesmann 85, der Epson-kompatibel ist und mit einem Centronics-Interface betrieben wird, tadellos.)

Als „letzte Rettung" finden die bedauernswerten Besitzer eines nicht-kompatiblen Druckers die Anleitung für einen Treiber (in Assembler) im Anhang des Handbuchs.

Wer Starpainter vom C 64 schon kennt, dem wird der Arbeitsbildschirm, der nach dem Laden erscheint, bestimmt bekannt vorkommen. Auch der Zeichenstift (durch einen Sprite erzeugt) ist derselbe geblieben.

Mit den Angaben auf den Menueleisten und den dazugehörigen Tasten lassen sich die Zeichengeschwindigkeit des grafischen Cursors einstellen (er liegt immer an der Spitze des Bleistifts), der gerade

FUNKTION IN MENUE-LEISTE AUFRUFEN

zu bearbeitende Bildschirmausschnitt auswählen, (acht gibts, zusammen ergeben sie einen Grafikbildschirm von 640 x 712 Pixels Auflösung, auf dem Drucker entspricht das einem DIN A-4-Blatt).

Aus einer Vielzahl von Füllmustern kann man — wenn's soweit ist — die passenden aussuchen, und letztendlich dem C 128 eigentlich sagen, was man jetzt zu tun gedenkt: freihändig per Joystick oder Maus zeichnen, Linien ziehen oder Rechtecke bzw. Kreise entwerfen (auch Ellipsen sind möglich!)

Eine weitere Funktion des Starpainter, die sich wohltuend von vergleichbaren Grafiktools abhebt, ist die Anzeige der gerade aktuellen horizontalen und

MASSSTABGERECHTES ZEICHNEN

vertikalen Position des Grafikcursors, sprich Zeichenstifts. Ein maßstabsgerechter Entwurf technischer Zeichnungen z.B. läßt sich so spielend verwirklichen. Man kann sich so eine eigene Straßenkarte entwerfen oder sein künftiges Haus bereits am Bildschirm bauen (auf einen Architekten wird man aber trotzdem nicht verzichten können...)

BEFEHLSEINGABE PER TASTATUR, JOYSTICK ODER MAUS

Tasten, die mit eigenen Funktionen belegt sind, erleichtern das Arbeiten mit diesem Programm. So läßt sich beispielsweise eine Befehlsübersicht jederzeit aufrufen („welche Taste muß man doch gleich wieder drükken, um Kreise zu zeichnen...?"), das Directory kann auf Wunsch angezeigt werden, HELP hat eine UNDO-Funktion, d.h., daß der zuletzt gewählte Modus wieder rückgängig gemacht wird, mit der ESC-Taste kommt man aus den Untermenues wieder heraus.

Grafiken sollten aussagefähig sein, in Bild und Wort. (Für Statistiken o.ä.) Starpainter 128 benützt dazu die deutsche Tastaturbelegung (sie gilt auch bei den guten Textverarbeitungsprogrammen für den 128er, wie SV-Text oder VIZAWRITE), somit

DEUTSCHE TASTATUR ZUR BESCHRIFTUNG DER GRAFIK

lassen sich alle Umlaute und Sonderzeichen auf dem Grafikbild darstellen. Wem das nicht ausreicht, der kann sich von Diskette andere Zeichensätze nachladen oder mit dem Zeichen-Editor, den das Programm ebenfalls bereit stellt, individuelle basteln, die man dann natürlich auch abspeichern kann.



Weil wir gerade beim Thema „Editor" sind: das ist nicht der einzige, den Starpainter 128 zur Verfügung stellt. Da ist beispielsweise der Sprite-Editor. Jetzt werden viele sagen, wieso denn, der C 128 bringt doch schon von Haus aus einen mit. Richtig, aber der ist mit dem Komfort des Sprite-Editors von Starpainter nicht zu vergleichen. Der kann einiges mehr: Das

SPRITE-EDITOR IN LUXUSAUSFÜHRUNG

kann nämlich einiges mehr: das Sprite läßt sich pixelweise in alle Richtungen verschieben, man kann es invertieren (Negativ-Darstellung), es läßt sich im Uhrzeigersinn jeweils um 90 Grad drehen, spiegeln oder ganz auf den Kopf stellen.

Der Zoom-Editor vergrößert bestimmte Ausschnitte (Shapes) einer Grafik, um diesen dann in Filigranmanier zu bearbeiten (wie durch ein Vergrößerungsglas) und zu guter Letzt bietet das Programm auch einen Muster-Editor, der die Mög-

EIGENE ZEICHEN ODER MUSTER MÖGLICH

(wir haben dieses bereits besprochen) zu entwerfen. Man könnte zwar annehmen, daß die von den Programmautoren entworfenen eigentlich ausreichen, um einem Bild unterschiedliche Schattierungen zur Belebung der ganzen Grafik zu geben, aber bitte, die Möglichkeit besteht!

Eines kann Starpainter 128 nämlich nicht: Irgendwelche Flächen mit den 16 Farben, die dem C 128 zur Verfügung stehen, auszufüllen. Lediglich die Farbe des Zeichenhintergrundes (= Zeichenblatt) und die der Striche und Linien (die der Bleistift malt) lassen sich beliebig anpassen.

DRUCKERFREUNDLICH

Was zunächst als Manko erscheint, stellt sich spätestens bei der Drukker-Ausgabe des Bildes als wohlüberlegt heraus. Verschiedene Farben auf dem Bildschirm sind noch lange nicht auf einem Schwarz-Weiß-Drucker (und um den geht es hier) verschieden. Der stellt nämlich jede Farbe mit derselben Bitkombination, wie sie die Zeichenfarbe (Color 1) hat, — egal ob rot, blau, grün — immer als schwarzes Pixel auf dem Papier dar. Im Multicolor-Modus ist das etwas anderes, weil hier die Bitbelegung für Zeichenfarbe, Multicolorfarbe 1 oder 2 anders aufgeteilt ist, so daß sich beim Ausdruck ein in gewisser Weise schattiertes Bild ergibt, aber Starpainter 128 ist nun mal ein Programm, das mit der hochauflösenden Grafik arbeitet (Hires-Modus).

1. wird damit ein feineres, genaueres Zeichnen erreicht und

2. ist es dem Starpainter nur in diesem Modus möglich, ein ganzes DIN-A-4-Blatt ausdrucken zu lassen (das ist dann nämlich die Zusammenfassung der einzelnen Bildschirme, die durch die Menuefunktion Fenster bereitgestellt werden).

SEINEN PREIS WERT

Auf einen Nenner gebracht: Starpainter 128 ist die konsequente Umsetzung des gleichen Programms für den Commodore 64, um es für seinen großen Bruder kompatibel zu machen. Es läßt sich einfach und sehr komfortabel bedienen, großmächtige Computerkenntnisse sind nicht erforderlich. (Wie man das Directory lädt und ein Programm startet, sollte man allerdings schon wissen!). Einen dicken Pluspunkt verdient das ausgezeichnete Handbuch, das sich sehr verständlich ausdrückt.

Gemessen an dem, was Starpainter 128 leistet, erscheint uns der Preis des Software-Pakets von DM 75,— nicht zu hoch. Erhältlich in jedem guten Fachgeschäft. (hb)

CP/M PLUS

Das dritte Betriebssystem

Mit großen Vorschußlorbeeren ausgestattet, macht doch kaum ein C128-Benutzer Gebrauch davon: CP/M, gesteuert vom Mikroprozessor Z 80.

Stellen wir zunächst die Kardinalsfrage: Wozu dient überhaupt ein Betriebssystem? Schließlich und endlich braucht einer den C128 nur einzuschalten, und schon meldet sich der Cursor mit READY und wartet auf die Eingaben. Und genau das ist der Kernpunkt. Ein Computer, egal, wie er heißen mag, besteht nicht nur aus Tastatur und Zentraleinheit (CPU), sondern aus mehreren Prozessoren wie etwa den Ein-/Ausgabe-Bausteinen, Videochips, Joystickanschluß, Diskettenstation, Userport usw.
Ein Betriebssystem kümmert sich nun darum, daß der Anwender oder Programmierer recht einfach auf diese genannte „Umgebung" (Peripherie) des Computers zugreifen kann. In der Regel ist es bei Homecomputern (wie der C128) so, daß jeder Rechner sein eigenes, meist zum Betrieb mit BASIC zugeschnittenes Betriebssystem mitbringt, das bereits fest eingebaut ist. (Drum ist es nach dem Einschalten auch präsent.)
Allerdings gibt's schon bei den diversen Homecomputer-Typen ein und derselben Hersteller-Firma erhebliche Unterschiede bei den BASIC-Dialekten, noch schlimmer wird es, wenn Sie beispielsweise ein auf dem C64 geschriebenes Maschinenprogramm im C128 betreiben möchten. Nicht, daß die Assemblercodes anders wären, nein, es sind genau dieselben – aber die Adressen der System-Routinen stimmen bis auf ganz wenige Ausnahmen nicht überein. Fazit: Ohne großmächtige Änderungen läßt sich kaum ein BASIC-Programm von einem C128 auf den C64 übertragen, und ein Assemblerprogramm auch nur dann, wenn Sie, grob gesagt, jede zweite Adresse ihrer Bestimmung entsprechend anpassen.

UNIVERSELL SOLLTE ES SEIN

CP/M dagegen wurde zu dem Zweck entwickelt, eine gemeinsame Programmierumgebung für die unterschiedlichsten Maschinen nicht nur desselben, sondern sogar verschiedener Hersteller zu ermöglichen. Daß dieser Versuch erfolgreich war, beweist die Tatsache, daß etwa 250 unterschiedliche Computertypen mit CP/M arbeiten können. Eine Standardisierung also.
Einige Grundkenntnisse dürfen wir voraussetzen: wie man erfährt, welche Dateien sich auf der Diskette befinden (DIR), wie kopiert wird (PIP) oder wie Dateien umbenannt werden. Auch

SUBMIT – DER STAPLER

wenn das Handbuch zur CP/M-Version 3.0 (wird beim Kauf eines C128 gratis mitgeliefert) für manche als „Witzblatt" empfunden wird, das alles steht da doch drin.
Drum möchten wir uns mit diesen allgemeinen Dingen auch nicht aufhalten, sondern Ihnen vielmehr Tips geben, wie Sie mit CP/M effektiv arbeiten können.
Zum Beispiel existiert da ein Dienstprogramm namens SUBMIT, das Stapelverarbeitung unter CP/M zuläßt. Wenn Sie sich aber unter Computer-„Kollegen" umhören, erfreut sich dieses Programm keineswegs der Beliebtheit, die seinem Leistungsumfang angemessen wäre.
Wie Sie wissen, werden CP/M-Befehle in der Kommando-Zeile (Command Line) editiert, also eingegeben und durch Druck auf die RETURN-Taste (sie wird ENTER genannt) zur Ausführung gebracht. Erst wenn so eine Ausführung zuende ist, können Sie den nächsten Befehl eintippen.
SUBMIT bietet nun die Möglichkeit, Kommandos hübsch der Reihe nach aus einer Textdatei zu lesen und abzuarbeiten.
Wenn's auch noch so schön klingt, beim ersten Ausprobieren wird's zum echten Ärgernis: Die Möglichkeit, eine Textdatei zu erstellen, ist sehr schlecht beschrieben. Es wird auf den Editor ED verwiesen. Die Arbeit mit diesem Programm verlangt in etwa die Sentimentalität eines Museumsdirektors, der seine Paßbilder mit minutenlangem Stillsitzen und Abbrennen von Magnesium erstellen will (statt ein Blitzlicht zu benutzen). Auch für aufgeschlossene Computer-Freaks ist ED ein echtes Rätselprogramm, auf das wir hier nicht näher eingehen wollen; vorzuziehen ist die Erstellung von Texten mit PIP. Natürlich ist auch die Arbeit damit umständlich, aber wesentlich leichter zu erlernen als ED. Außerdem nehmen wir doch an, daß jeder, der ernsthaft mit CP/M arbeiten will, sich über kurz oder lang einen Editor oder ein Textverarbeitungsprogramm anschafft, um etwas umfangreichere Texte zu erstellen.
PIP ist eigentlich ein Dienstprogramm zum Kopieren von Dateien unter CP/M.
Neben den tatsächlichen Dateien auf Diskette kennt CP/M aber auch sogenannte „logische" Dateien wie Tastatur, Drukker oder die V24-Schnittstelle.
Manche dieser „Dateien" eignen sich nur zum Lesen, wie's beispielsweise bei der Tastatur der Fall ist.
Demnach ist unser Wunsch,

PIP ALS TEXTEDITOR

per Tastendruck eine Textdatei auf Diskette zu erzeugen, nichts anderes als ein Kopieren von der Tastatur auf eine Diskettendatei mit einem von uns gewählten Dateinamen.
Rufen wir also PIP auf: A> PIP, das Programm meldet sich mit dem bekannten Sternchen „*" und wartet auf eine Anweisung. Nehmen wir mal an, unser erster Versuch soll TEST.TXT heißen, die logische Datei, die wir lesen wollen, ist die Tastatur, im CP/M-Sprachgebrauch „CON" genannt (von „Console"). Die Anweisung an PIP sieht dann so aus:

*A:TEXT.TXT=CON:

(RETURN/ENTER-Taste drücken)
(Den Stern am Zeilenanfang nicht nochmals eintippen, das stammt vom Aufruf von PIP!)
Der Cursor rückt in die nächste Zeile. Nun könnten wir eintippen:
"DIES IST EIN TESTTEXT."
Drücken wir jetzt ENTER, wandert der Cursor zum Zeilenbeginn. So war das aber nicht gedacht, schließlich wollten wir ja mehrere Zeilen erstellen.
Das Rätsel ist jedoch schnell gelöst: RETURN oder ENTER hat den Tastencode 13 (wie in BASIC); physikalisch bewirkt unter CP/M dieses Zeichen nur einen Wagenrücklauf (CR = Carriage

Return), nicht aber noch zusätzlich einen Zeilenvorschub (LF = Line Feed). (In BASIC ist auch der dabei.)
Diesen „Line Feed", Code 10, können wir durch CTRL-J erzwingen; nun steht der Cursor da, wo wir ihn haben möchten: Am Anfang der nächsten Zeile.
Weitere wichtige Steuerzeichen mit Hilfe der CONTROL-Taste finden Sie in einer Tabelle zu diesem Artikel, vor allem, wenn sich einer vertippt hat, muß er ja mit dem Cursor auch innerhalb der Zeilen wandern können (wie beim komfortablen Bildschirmeditor des C128 bei BASIC-Betrieb).
Nachdem wir so einen kleinen Text eingegeben haben, sollte uns PIP den natürlich auch auf Diskette schreiben. Seien Sie bitte aber nicht zu übereifrig: Wenn Sie mit CTRL-C den Editiermodus verlassen haben, werden Sie vergeblich nach der Datei TEXT. TXT suchen. PIP glaubt nun, Sie hätten einen Fehler gemacht und das Programm dadurch mit CTRL-C abgebrochen. Dann einen (falsch) editierten Text abzuspeichern, ist CP/M doch zu dumm.

ZUM ABSCHLUSS CTRL-Z DRÜCKEN

Da die Console (CON:) als Textdatei angesprochen wird, sucht PIP stets nach dem Zeichen EOF (End of File = File-Ende), das unter CP/M den Code 26 besitzt. Erst wenn dieser Code erkannt wird, schließt CP/M die Datei CON: und Zeichen für Zeichen einschließlich des EOF wird als neue Datei TEXT.TXT auf die Diskette geschrieben. Dieses EOF erreichen Sie mit CTRL-Z.
Also, daran denken: Einen mit PIP erstellten Text beliebiger Länge nur mit CTRL-Z abschließen und PIP verlassen. Falls Sie nicht glauben, daß es geklappt hat, tippen Sie zur Kontrolle ein

TYPE TEXT.TXT
(ENTER-Taste)

Sie können erkennen, daß PIP auch die CTRL-Codes mit in die Datei übernimmt; TYPE ist so etwas wie ein LIST-Befehl; bei der Bildschirmausgabe werden diese Codes jedoch wieder neu interpretiert, so daß Sie den Text so sehen, wie Sie ihn haben wollten.

RAN AN CP/M!

Mit diesen ersten „fundierten" Kenntnissen möchten wir uns nun endlich daran machen, die erste SUBMIT-Befehlsdatei zu erstellen. Um bei Aufruf von SUBMIT nicht stets diese sechs Buchstaben eintippen zu müssen – wer arbeitet schon gerne unnötig – legen wir uns eine Kopie vom File SUBMIT.COM namens DO.COM an.
Jetzt stellt sich der Aufruf als „DO" dar, das sind immerhin vier Buchstaben weniger, bei 1000 Aufrufen sind das 4000 Tastendrucke weniger, bei 10000 dann ... (Scherz beiseite).
Das Kopieren erledigt wieder:
PIP (ENTER)
*DO.COM=SUBMIT.
COM (ENTER)
*
Schreiben wir uns doch mal eine Befehlsdatei, die den freien Diskettenplatz und die auf Disk vorhandenen Dateien ausgibt:
*INFO.SUB=CON:
(ENTER)
(Achtung: Niemals den Filenamen-Zusatz „SUB" vergessen, sonst kommt SUBMIT/DO nicht damit zurecht!)
Der Eingabetext für die zu bastelnde Datei sieht dann so aus:

SHOW.A:
DIR.A:

Nach CTRL-Z können wir PIP verlassen und mit DO INFO (den Zusatz „SUB" brauchen Sie jetzt wiederum nicht mehr angeben, denn nun liegt er ja bereits vor und wird automatisch von CP/M angehängt) wird unser erstes Programm, ein sogenanntes „BATCH"-Programm ausgeführt. Sollte CP/M Fehlermeldungen ausgeben, so ist ein solcher leicht zu finden: Bevor SUBMIT einen Befehl aus der Datei ausführt, wird die entsprechende Zeile auf dem Bildschirm ausgegeben, Tippfehler lassen sich dadurch leicht erkennen. Maschinenintern wird bei der Abarbeitung dieser Befehlsdatei eine Zwischendatei mit dem Zusatz .$$$ erzeugt; erst aus dieser werden die eigentlichen Befehle ausgelesen und ausgeführt.
Warum so umständlich? Das liegt an einem weiteren Vorteil von SUBMIT. Es besteht die Möglich-

BATCH-DATEIEN

keit, automatisch Platzhalter zu ersetzen.
Platzhalter sind dazu da, Wörter, die beim Aufruf von SUBMIT/DO in der Kommandozeile hinter dem Namen der Befehlsdatei stehen, in dieselbe einzufügen – und dazu braucht CP/M eben die Zwischendatei, um die „echte" Batch-Datei nicht zu zerstören. Stellen, an denen Ersetzungen vorgenommen werden sollen, werden mit dem Dollarzeichen und einer Zahl angegeben: Die Zahl bedeutet hierbei die Nummer, an deren Stelle in der Parameterliste das betreffende Argument erwartet wird. Klingt kompliziert, stimmt's? Ist es aber gar nicht.
Nehmen wir einmal an, wir haben DO mit folgender Zeile aufgerufen:

DO.INFO*.COM*.DTA

DO ruft SUBMIT auf. (Wir haben es ja unter diesem neuen Namen kopiert.) INFO ist die Datei, in der die Befehle stehen, *.COM ist Argument 1, *.DTA ist Argument 2. Sollte in der Datei INFO. SUB nun irgendwo "$1" stehen, ersetzt DO *jede* Stelle, an der $1 steht, durch das vorhandene Argument "*.COM". Entsprechend werden alle $2-Platzhalter durch "*.DTA" ersetzt. Ist kein Argument angegeben, werden die Platzhalter

WICHTIGE EDITIER- UND STEUERZEICHEN BEI CP/M

Tastenbefehl	Funktion
CTRL-A	Cursor ein Zeichen nach links
CTRL-C	Programmabbruch
CTRL-E	Zeilenvorschub ohne ENTER-Taste (Line Feed)
CTRL-F	Cursor ein Zeichen nach rechts
CTRL-G	Löscht das Zeichen, auf dem der Cursor steht
CTRL-H	Löscht das Zeichen links neben dem Cursor
CTRL-M	Enter-Funktion
CTRL-W	Gesamte letzte Zeile wird wiederholt
CTRL-Z	EOF, Dateiendekennzeichen (End of File

durch Leerzeichen ersetzt. Das nämliche gilt für den Fall, daß weniger Argumente angegeben werden, als Platzhalter in der Befehlsdatei angefordert wurden: auch hier werden einfach Leerzeichen eingesetzt. Um die Funktion auszutesten, schreiben wir uns am besten eine neue „Batch"-Datei namens INFO2.SUB:

```
SHOW.A:
DIR.A:$1
TYPE $2
```

Wie sieht es nun aus, wenn sie abgearbeitet wird?
Aufruf:
DO INFO2*.TXT
INFO2.SUB
Zwischendatei:
SHOW.A:
DIR.A:*.TXT
TYPE INFO2.SUB
Resultat:
Diskettenplatz auf A: wird angezeigt, alle Dateien mit Zusatz .TXT werden angezeigt und der Inhalt von INFO2.SUB auf dem Bildschirm ausgedruckt. Der Aufruf DO INFO2 verhält sich dagegen genau wie das Programm DO INFO: Da DIR ohne Parameter bleibt, werden alle Dateien angezeigt, und TYPE ohne Argument macht überhaupt nichts. Die mögliche Anzahl der Argumente/Platzhalter ist mit insgesamt 10 übrigens völlig ausreichend, bedenken Sie dabei jedoch, daß der Computer bei „Null" zu zählen beginnt: S0 bis S9.

CP/M UND PASSWORT

Oft wurde die Frage gestellt, welchen praktischen Sinn die von CP/M angebotenen Möglichkeiten der Definition von Paßwörtern, geschützten Dateien, Benutzerbereichen usw. haben sollen. Ganz ehrlich: Beim 128 PC überhaupt keinen! Alle diese Utilities wurden entwickelt, als CP/M als ein für die damalige Zeit sehr leistungsfähiges Betriebssystem überwiegend im geschäftlichen Bereich eingesetzt wurde. Ein einziger Computer stand da für viele Benutzer zum Beispiel in einer Firma zur Verfügung, die Massenspeicher waren Winchester-Laufwerke oder Festplatten mit 20 und mehr Megabyte Kapazität. Um diese große Datenmenge zu verwalten, war es erforderlich, den zur Verfügung stehenden Platz für mehrere Benutzer aufzuteilen, damit jeder so arbeiten konnte, als habe er seine eigene Festplatte mit vielleicht nur einem oder zwei Megabyte. Da sollte natürlich der Kollege aus einer anderen Abteilung oder eines fremden Ressorts tunlichst seine Nase aus den Dateien lassen.
Für den Privatbereich zu Hause und den Betrieb einer oder zwei Diskettenstationen brauchen keine Paßwörter oder Benutzerbereiche festgelegt werden; schneller ist eine Diskette gewechselt und unter Verschluß gebracht als ein Paßwort gesetzt.

RMAC UND LINK

CP/M Plus bietet auch Programmierwerkzeug an: Da gibt es den Macro-Assembler RMAC und den dazugehörigen Linker LINK. Wer als Programmierer mit Compilersprachen arbeitet (Pascal, C) wird feststellen, daß auch einem Compiler nicht immer alles möglich ist. Manch einer möchte sich eine Assembler-Routine auch selbst schreiben. Dann freut sich der um so mehr, ein entsprechendes Werkzeug zur Hand zu haben, das dazu noch so weit verbreitet ist, daß die meisten Compiler die Option besitzen, mit RMAC geschriebene Routinen direkt mit einbinden zu können.
Die Bedienung ist wirklich einfach. Tippen Sie doch das kleine Beispielprogramm ab, es ist in der 8080-Assemblersprache geschrieben; Spezialisten werden wissen, daß der im 128 PC eingebaute Z80-Prozessor kompatibel zum INTEL 8080 ist, aber noch ein paar Befehle mehr versteht. Dieses kleine Programm muß jetzt in Maschinensprache, den „Object-Code", umgesetzt werden. Dazu rufen wir den Assembler RMAC auf. RMAC erzeugt noch keine richtige Maschinensprache, sondern eine Datei, in der eventuelle externe Adressen, Makros usw. freigelassen werden. So eine Datei trägt den Zusatz ".REL" für „relativierbar". Daneben wird ein Übersetzungsprotokoll mit dem erzeugten Code erstellt; in der Regel wird so ein Protokoll ausgedruckt, daher enthält es den Zusatz ".PRN" (Printer).
Die Namen für die Dateien können wir explizit vorschreiben:
RMAC FILE1, FILE2= FILE3 erzeugt FILE1. REL und FILE2.PRN aus der Datei mit Assemblercode FILE3.ASM, wobei .ASM als Zusatz vorgeschrieben ist. Übersetzen wir also unser Programm:
TEST.ASM
RMAC TEST,PFILE =TEST
Wenn Sie sich nicht vertippt haben, wird ordnungsgemäß übersetzt und kein Fehler gemeldet. Falls Sie keine PRN-Datei brauchen, dann lassen Sie einfach diesen Parameter weg.
RMAC REFILE,=TEST
Nun kommt noch das „Binden" (Linken). Da wir nur eine REL-Datei linken müssen, genügt:
LINK TEST
Mit DIR TEST.* vergewissern wir uns, daß ein Programm TEST.COM auf Diskette existiert; mit TEST wird es aufgerufen. Und schon haben Sie Ihr erstes Assembler-Programm unter CP/M geschrieben.
Wer tiefer in CP/M einsteigen möchte: Eine empfehlenswerte Anschaffung, da auch in deutscher Übersetzung vorliegend, ist Rodney Zak's „Einführung in CP/M" (Sybex-Verlag, Düsseldorf).
Last not least kommt schließlich noch das Programm LIB.COM ins Spiel, mit dessen Hilfe Sie bequem auch große Mengen bereits übersetzter Assembler-Programme verwalten können (LIB kommt von „Library"). Auch hier möchten wir Sie auf weiterführende Literatur verweisen (CP/M User's Guide usw.).

CP/M UND HOCHSPRACHEN

Programmieren in Assembler ist zwar die hohe Kunst, doch existiert für CP/M eine derartige Fülle von Compilern, daß selbst für den eingefleischten „Puristen" und Geschwindigkeitsfanatiker 95% der Software von einer Hochsprache zu realisieren ist.
Auch sollten Sie beachten, daß Programme, die in Assembler erstellt wurden, beim Um- oder Aufstieg beispielsweise zum Amiga nur noch fürs Archiv taugen; Pascal-Programme dagegen sind meist innerhalb weniger Stunden übertragen, bei strikter Einhaltung der Konventionen sogar innerhalb weniger Minuten.

BASIC

Die wichtigsten CP/M-Vertreter dieser Sprache sind CBASIC und MBASIC; letzteres ist recht preiswert (ca. 200 Mark) und bietet neben einem Compiler auch noch einen Interpreter sowie einen Makro-Assembler, M80, den Linker L80 und ein Bibliotheks-Utility-Programm namens LIB80. Vorteil: Sie können mit Z80-Maschinensprache arbeiten, außerdem ist MBASIC ein überaus „kompatibles" BASIC, das auch unter MS-DOS (das gebräuchlichste Betriebssystem der großen PC-Rechner) sowie

mit Einschränkungen auch dem AmigaBASIC gleicht.

PASCAL
Unbestrittene Nummer eins ist hier Turbo Pascal; komfortabel in der Programmentwicklung, billig zu erstehen und auch sehr portabilitätsfreundlich, also auf andere Computer übertragbar, da es den Pascal-Standard weitestgehend einhält. Preis: Bei knapp 200 Mark.
Nicht zu verachten ist aber auch Nevada Pascal, insbesondere, wenn Sie viele Assembler einbinden möchten. In der „Public-Domain"-Version (= frei verfügbare Software für jeden, SIG/M Nr. 82 als „JRT-Pascal") kostet es mit Assembler, Linker, Debugger (=„Fehler-Entwanzer") und vielen Hilfsprogrammen ab zehn Mark und entspricht ebenfalls dem Standard. Bei dem Preis ist allerdings kein Handbuch und kein Editor dabei . . .

FORTRAN/COBOL
Diese Sprachen unter CP/M sind wohl in der Hauptsache für denjenigen interessant, der sie ohnehin schon beherrscht. Von Nevada gibt es beide zu einem Preis von etwa 100 Mark.

FORTH
Für alle Fans dieser Computersprache dürfte ein dickes Software-Paket ebenfalls aus der „Public Domain" (SIG/M Nr. 204, FORTH83) sein: Multitasking-Simulation (der Computer kann mehrere Aufgaben zur gleichen Zeit ausführen), rieisger Sprachumfang. Fast 600 KByte Daten, Infos, Programme, Dokumentationen – kurz, ein Paket, das alle Wünsche erfüllt. Es ist ebenso wie JRT-Pascal schon ab zehn Mark erhältlich.

„C"
Wenn Sie die COMMODORE DISC 17 aufmerksam gelesen haben, wissen Sie bereits um die grandiosen Vorzüge dieser Sprache. dBase III und UNIX, Word 2000, große Teile des Amiga-Betriebssystems und vieles andere ist in „C" programmiert. Vornehmlich mit zwei Versionen sollten Sie sich befreunden: BDS-C und C80.
Das Erstgenannte ist in den USA sehr weit verbreitet; eine stattliche Anzahl Programme, auch aus der „Public Domain", ist mit diesem Compiler erstellt worden. Leider ist es in Europa bislang kaum zu bekommen, außerdem besitzt es recht eigenwillige Funktionen.

BEISPIELPROGRAMM FÜR ASSEMBLER RMAC/LINK

```
CPM    EQU 05H; Schnittstelle
       ORG 0100H; 1. Adresse
       LXI D,TEXT; LD DE,TEXT
       MVI C,0;  LD C,9
       CALLCPM; Stringausgabe
       RET

TEXT:  DB 'Hier meldet sich'
       DB 'der C128 mit dem'
       DB 'ersten Assembler'
       DB 'programm.'
       DB 0AH,0DH; Linefeed/Carr.Ret.
       DB 'Programmierer:'
       DB '(Ihr Name!)'
       DB 0AH,0DH
       DB '$', Ende des Strings
       END
```

C80 ist nicht gerade billig, läßt aber fast alle Tricks zu, wegen denen „C" so beliebt ist. Große Bibliothek und ausführliche Dokumentation machen den Preis (278 Mark) wieder wett; da der ANSI-Standard weitgehend erfüllt ist, können wir es Ihnen nur empfehlen.
Ferner gibt's noch ein SMALL C in der Public Domain; zum Kennenlernen ganz nett, aber im Sprachumfang so eingeschränkt, daß Sie größere Programmpakete nicht verwenden können. Doch es ist ideal zum „Reinschmecken", bevor Sie eine größere Ausgabe für ein komfortables „C"-Programm tätigen, um dann festzustellen, daß Ihnen diese Sprache doch nicht so behagt.
Übrigens: Die „professionelle" Version, die unter demselben Namen angeboten wird, bietet außer einem Handbuch und einem Editor auch nicht mehr. Nur kostet es viel mehr als die volkstümliche Public-Domain-Fassung.
Neben tausend Speziallösungen insbesondere im geschäftlichen und meßtechnischen Bereich sind vor allem drei Programme mit sehr allgemeinen Anwendungsmöglichkeiten zu nennen: dBase II, WordStar und Multiplan.
dBase II, unter CP/M und MS-DOS erhältlich, kostete noch vor einiger Zeit etwa 1000 Mark, da aber die Entwicklungskosten bei weitem schon gedeckt waren, konnte der Hersteller (Ashton Tate) überzeugt werden, daß eine Unzahl von 128PC-Besitzern in Europa auch Interesse an diesem Programm zeigen könnte, was natürlich nur eine Frage des Preises sei. Ergebnis: dBase gibt's inzwischen für etwa 200 Mark.
Dafür erhalten Sie ein ausgereiftes Datenbanksystem, das dem Benutzer jede Freiheit läßt, eine eigene Sprache bereitstellt, Programmierung erlaubt und vor allem mit sehr umfangreichen Erklärungen und Dokumentationen ausgeliefert wird, kurz: *die* Datenbank für CP/M-Computer.
Dem Textverarbeitungsprogramm WordStar kann getrost nachgesagt werden, daß es der Wegbereiter für alle nachfolgenden bildschirm-orientierten Editoren (sowas ist nämlich ein Textverarbeitungsprogramm) im weitesten Sinne war. WordStar glänzt weniger durch seine Geschwindigkeit oder seinen Bedienungskomfort, sondern vielmehr durch seine Verbreitung und seinen Bekanntheitsgrad. Voll ausnutzen kann das Programm, wer „Mailmerging" betreiben, Drukker anpassen, Formatieren, Formulare drukken und dergleichen mehr möchte. Zum Schreiben von einfachen Briefen, Programmen im Quellcode oder simplen Zeitungsartikeln, wie etwa den, den Sie gerade lesen, genügt vollauf der in Turbo-Pascal (Vertrieb: Firma Heimsoeth, München) eingebaute Editor, der in seinen Funktionen sowieso kompatibel zu WordStar ist.

MULTIPLAN – DAS „SPREADSHEET"
Multiplan (von Microsoft) hat gewisse Gemeinsamkeiten mit dBase: Es ist seit geraumer Zeit zu einem weitaus günstigeren Preis zu haben als ehedem; es ist die „große alte Dame" der Sparte; alles Vergleichbare ist mehr oder weniger „nachprogrammiert". Und: es ist absolut ausgereift.
Wir können hier unmöglich alle Funktionen beschreiben, das macht das 350seitige Handbuch dazu recht ordentlich, zudem gibt's bei den be-

kannten Fachbuchverlagen sehr gute ergänzende Bücher (Data Becker und Sybex, beide Düsseldorf, Markt und Technik, Haar).
Die in den vergangenen Jahren vor allem für MS-DOS sehr oft verkauften Pakete „Eines für alle“, Kalkulation, Textsystem, Datenbank, eventuell Modemsteuerung, automatische Verwaltung der Festplatte vom Programm aus und dergleichen sind zum Teil erst durch die rapide gesunkenen Preise für die RAM-Speicher möglich geworden. Solche Pakete arbeiten mit Programmgrößen von mehreren 100 KByte; für CP/M undenkbar, da es ja stets nur 64 KByte gleichzeitig ansprechen kann, das Los jedes Acht-Bit-Computers. Und genau für diese „Bit“-Zwerge, zu denen auch der C128 gehört, wurde ja CP/M entwickelt. Für die großen Personalcomputer, die schon 16 Bits adressieren können, wurde aus CP/M dann MS-DOS entwickelt. Doch brauchen wir CP/M-ler den Kopf nicht hängen lassen: Die drei Klassiker dBase, WordStar und Multiplan wurden mit der Zeit so bedeutsam, daß es sich keiner der drei verschiedenen und voneinander unabhängigen Hersteller mehr erlauben konnte, „Insellösungen“ zu schaffen. Das Ergebnis: dBase-Dateien lassen sich mit WordStar editieren oder zum „Mailmerging“ lesen. Multiplan-Tabellen können von den beiden anderen direkt verarbeitet werden, dafür lassen sich Recherchen aus dBase direkt mit Multiplan bearbeiten, die Ergebnisse in WordStar-Reports einlesen und ausdrucken usw. Verglichen mit den Preisen für MS-DOS-Lösungen wie etwa Framework, das im Prinzip auch nicht mehr kann – zumindest nicht in dem Sinne, daß der Privatanwender oder kleinere Betrieb diese Möglichkeiten auch ausnutzen könnte –, kommen Sie mit insgesamt etwa 600 Mark für alle drei Programme um gut 1000 Mark besser weg als der Besitzer eines „echten“ MS-DOS-Personalcomputers.
Angesichts der immer billiger werdenden 16-Bit-Rechner sowohl unter den PCs als auch der „68000er“-Familie (Atari ST, Amiga, Apple) muß sich der Besitzer eines 128 PC durchaus die Frage stellen: Was wird aus CP/M? Betrachten wir es mal nüchtern und ohne rosarote Brille: Zukunft hat das gute, alte 8-Bit-CP/M wohl keine mehr. Programmpakete wie dBase werden nicht mehr entwickelt werden: Die Programmierer dieser Firmen sind voll dabei, diese alten Programme 1:1 auf die Rechner der neuen Generation umzusetzen, auch wenn das den Eindruck eines Anachronismus vermittelt. Gute Ideen sind eben zeitlos. Wir als CP/M-Anwender können uns jedoch auf etwas berufen, das die bestaunten 32-Bit-Computer in absehbarer Zeit noch nicht haben werden: ein sehr ausgereiftes Standard-Betriebssystem, eine ruhmreiche Vergangenheit und eine mit einer Million CP/M-Geräten allein in Deutschland (natürlich sind das nicht nur Commodore-Rechner) nicht wegzudiskutierende Zahl von Anwendern.
Die Hardware-Produzenten, das ist deren Geschäft und auch ihr gutes Recht, versuchen uns einzureden, wir bräuchten jedes Jahr einen neuen Computer.
Die Software-Häuser hingegen bemühen sich, uns einzureden, daß genau ihre Programme auf genau unserem Computer gut seien.
Dies ist die Gegenwart von CP/M: eine Gegenwart, die es uns leicht fallen läßt, ein „veraltetes“, ein „kleines“ Betriebssystem zu benutzen.

LM□

DATENTRANSFER

Mit der Buschtrommel begann es

Von jeher war der Mensch auf Informationen und Nachrichtenaustausch erpicht. Dabei spielte natürlich die Geschwindigkeit des Austausches eine große Rolle. Zuerst wurden Kuriere losgeschickt, später dann Rauch- und Feuerzeichen verwendet. Die Urwaldtrommler könnte man als erste akustische Nachrichtenübermittler bezeichnen. Als das technische Zeitalter einsetzte, wurde der Informationsaustausch immer schneller und bequemer. Durch die Erfindung des Telefons gelang es, Informationen in großer Menge zu übermitteln. In unserer heutigen Zeit übernehmen diese Aufgaben immer mehr die Computer, zuerst im professionellen und zunehmend im privaten Bereich.

Es gibt viele Möglichkeiten, Texte von einem Computer zum anderen zu schicken. Den einfachsten Weg bietet die Telekommunikation.
Benötigt werden eine Telefonanlage, für jeden Computer je ein Akustikkoppler, das abgedruckte Programm für Ihren Rechner und ein Terminal-Programm für die Gegenstelle.
Bei einem Telefongespräch wird aus dem Schalldruck des gesprochenen Wortes eine Spannung mit wechselnder Frequenz erzeugt. Diese wird über die Leitung übertragen und Leitungsverluste durch Verstärker wieder ausgeglichen. Beim Empfänger wird diese Wechselspannung wieder in einen hörbaren Schalldruck zurückverwandelt.
Digitale Signale, die bei unserer Datenfernübertragung vorkommen, müssen in akustische Signale umgewandelt werden. Nur dann können sie auch über die Leitung gesendet werden. Beim Empfänger angekommen, werden diese akustischen Signale wieder in digitale umgesetzt. Die Geräte, die eine solche Umsetzung bewerkstelligen, nennt man Modems oder Akustikkoppler.

Die Übertragungs-Geschwindigkeit beträgt 300 Baud im Vollduplexverfahren. Das heißt, es werden in einem Datenkanal gleichzeitig Daten gesendet und empfangen. Dieses Verfahren wurde in Europa genormt und trägt die Bezeichnung CCITT (Comite Consultative International Telegraphique et Telefonique).
Es gibt zwei verschiedene CCITT-Normen. Zum einen die CCITT-V.21-Norm, die zwei gleich große Datenkanäle besitzt. Damit lassen sich aber nur Geschwindigkeiten von 300 Baud oder weniger erreichen.

Wollen Sie aber mit höheren Übertragungsarten arbeiten, so können die Daten nicht mehr gleichzeitig empfangen und gesendet werden. Das liegt daran, daß in einem Telefonkanal keine zwei Datenkanäle mit sicherem Störabstand untergebracht werden können. Sie müssen also auf das Halb-Duplexverfahren, die

CCITT-V.23-Norm, umschalten.
Hierbei werden die Daten entweder gesendet oder empfangen. Das Ende einer Übertragung wird als Schlußzeichen oder über einen langsamen Datenkanal übermittelt. Im Hauptkanal werden Daten mit 1200 bps (Bits per Second) übertragen, ein Zusatzkanal mit 75 bps wird für die Steuerung der Datenübertragung verwendet.

DIE RS-232-SCHNITTSTELLE

Den Datenfluß zwischen Computer und anderen Peripherie-Geräten wie Drucker oder Akustikkoppler regelt die Schnittstelle RS-232, in Europa auch V24 genannt. Sie ist am geeignetsten für unsere Übertragung per Telefon. Die Übertragung erfolgt immer seriell, ein paralleler Transfer über Telefon ist nicht möglich. Bei der seriellen Übertragung werden die acht Bits eines Bytes einzeln übertragen. Dadurch wird zwar die Geschwindigkeit geringer, aber Sie kommen mit weniger Leitungen aus.
Die Schnittstelle am Commodore ist durch ein Modul erhältlich, beziehungsweise ein Kabel, das auf der Commodore-Seite einen Userportstekker und auf der anderen einen 25poligen Stecker (DB25) aufweist. Diese Kabel gibt es in der Regel zu kaufen, sie nehmen auch die erforderliche Pegelumwandlung ±12 Volt vor. Sollten Sie aber vorhaben, dieses Kabel selbst zu basteln, so entnehmen Sie bitte die Pin-Belegung der beiden Stekker unserer Grafik.
Achten Sie aber bei der Verwendung des Userports als Eingang unbedingt auf die Eingangsspannung. Sie darf nur in einem Bereich von null bis fünf Volt liegen.
Soll der Userport als Ausgang dienen, halten die

Term.128

```
10 rem ==========================  <cm>
20 rem parameter 8 n 1              <hi>
30 rem ==========================  <mm>
100 open 2,2,3,chr$(6):printchr$(1
2)+chr$(14):printchr$(147)          <in>
110 dim g%(256),h%(256)             <nm>
120 for s=0to255:h%(s)=s:next       <dh>
130 fors=32to64:h%(s)=s:next        <mk>
140 fors=65to90:h%(s)=s+32:next     <ad>
150 fors=97to122:h%(s)=s-32:next    <hk>
160 h%(20)=8                        <pf>
170 fors=193 to218:l=s-128:h%(s)=l
:next                               <gd>
180 fors=0to255                     <mi>
190 l=h%(s)                         <mg>
200 ifl<>0theng%(l)=s               <pm>
210 next                            <ch>
220 print"Uebertragung-Start"       <ie>
230 get#2,a$                        <oa>
240 ifa$=""orst<>0then270           <nf>
250 ifa$=chr$(10)then230            <fh>
260 print chr$(g%(asc(a$)));        <ge>
270 geta$                           <kf>
280 ifa$=chr$(95)then310            <ja>
290 ifa$<>""thenprint#2,chr$(h%(as
c(a$)));                            <im>
300 goto 230                        <hm>
310 print:print"Filename:";:poke21
,1:inputb$                          <pn>
319 poke21,0:print:open8,8,8,b$+",
s,r"                                <jm>
320 get#8,a$:su=st:print#2,a$;:pri
nta$;                               <id>
330 ifa$<>""thenprint#2,chr$(h%(as
c(a$)));                            <od>
340 if su=0then320                   <ad>
350 close8:goto230                  <nn>
360 rem =p r o g r a m m e n d e=   <mo>
```

Ausgänge nur die Belastung eines TTL-Einganges aus. Es empfiehlt sich in jedem Fall, eine Pufferstufe einzubauen.

OHNE TERMINAL-PROGRAMM GEHT NICHTS

Das nächste wichtige Handwerkszeug zur Übertragung von Daten ist das Terminalprogramm. Wollen Sie zwei Commodore-Rechner verbinden, reicht das Grundprogramm des zweiten Listings.

Unser Beispiel zeigt Ihnen ein kleines Terminalprogramm für die Verbindung zwischen Commodore und PC. In Zeile 100 wird die RS-232-Schnittstelle geöffnet, der Bildschirm gelöscht und auf Kleinschrift umgeschaltet. Den OPEN-Befehl sind Sie von der Floppy gewohnt. Die Geräteadresse der RS-232-Schnittstelle lautet immer 2. Danach folgt die Parametereinstellung (Baud, Stopbits, Parität) mittels zweier CHR$-Anweisungen. In unserem Beispiel reicht für eine Übertragung mit 300 Baud, acht Bits, Vollduplex und keiner Paritätsprüfung der Befehl Chr$(6).
Für eine Übertragung mit 300 Baud, sieben Bits, Vollduplex und keiner Parität müßte der Befehl OPEN2,2,0,Chr$(38)+Chr$(224) lauten.
Der Commodore legt sich nach dem OPEN-Befehl zwei Puffer mit je 256 Bytes an. Im einen werden die empfangenen und im anderen die zu sendenden Daten bis zur Weiterverarbeitung aufbewahrt. Die Geschwindigkeit, mit der die Daten an die Schnittstelle geschickt werden, ist immer konstant. Daher ist es bei einer größeren Geschwindigkeit nötig, die Daten in einem Puffer zwischenzulagern.
Das gleiche geschieht mit jenen Daten, die bereits im Rechner angekommen sind, aber noch nicht weiterverarbeitet wurden. In Zeile 110 werden dann zwei Variablen zur Datenkonvertierung initialisiert. G% steht für die Empfangsdaten und H% für die Sendedaten.
Da bei allen Commodore-Heimcomputern der ASCII-Code umgekehrt zum Standard-ASCII-Code verschlüsselt ist, muß eine Codewandlung vorgenommen werden. Dies geschieht in den Zeilen 130 bis 170. Die Codewandlung betrifft aber nur die Klein- und Großbuchstaben. Bei Satzzeichen und ähnlichem entspricht der Commodore-Code dem des Standards.
Wird eine Taste gedrückt und liegt der ASCII-Wert des eingelesenen Zeichens im Bereich zwischen 64 und 91, dann ist es ein Kleinbuchstabe. Soll dieses Zeichen gesendet werden, so muß der Wert 32 addiert werden (Zeile 140). Das DELETE-Zeichen (20) wird in Zeile 160 zum Standard-BACKSPACE-Zeichen (8) umgewandelt.
Beim Empfang eines Zeichens wird vom ASCII-Wert 32 abgezogen, wenn

es ein Kleinbuchstabe war (Zeile 150).
Vergleichen Sie dazu die Tabelle der Standard-Codes mit der aus Ihrem Handbuch.

VON DER THEORIE ZUR PRAXIS

Mit unserem kleinen Terminalprogramm, das Sie abgetippt und auch hoffentlich abgespeichert haben, können Sie bereits mit Mailboxen oder anderen Computern kommunizieren.
Als erstes verbinden Sie Ihren Rechner mit dem Akustikkoppler, schalten ihn ein und stellen den Wahlschalter auf ORIG (originate). Sollten Sie nicht mit einer Mailbox, sondern mit einem anderen Computer kommunizieren wollen, so muß dessen Akustikkoppler auf ANSW (answer) geschaltet sein.
Schalten Sie Ihren Computer ein und starten Sie das abgetippte Terminalprogramm.
Plazieren Sie Ihr Telefon so, daß Sie den Telefonhörer bequem mit dem Koppler verbinden können. Achten Sie unbedingt auf die richtige Lage des Hörers. Als nächstes wählen Sie die Nummer des gewünschten „Ansprechpartners“.
Ertönt nach dem Rufzeichen ein hoher Pfeifton, ist die Gegenstelle zum Datenaustausch bereit.
Wenn Sie mit einer Mailbox verbunden sind, so bestätigen Sie mit der RETURN-Taste, daß auch Sie bereit zum Datenaustausch sind. Der weitere Verlauf ergibt sich durch das Mailboxmenü. Da alle Mailboxen mit unterschiedlicher Menü-Führung arbeiten, gibt es keine einheitlichen Wegweiser durch die Systeme. Sie müssen sich selbst einarbeiten.
Wenn Sie ein vorbereitetes sequentielles File senden wollen, so drücken Sie die „Pfeil-nach-links-

Term

```
10 rem -------------------------          <cm>
20 rem parameter 7 n 1                    <he>
30 rem -------------------------          <mm>
100 open 2,2,0,chr$(38)+chr$(224)         <nc>
110 printchr$(147);chr$(14)               <cb>
130 geta$: ifa$=""then170                 <mc>
135 printa$;                              <mn>
140 a=asc(a$):ifa<91anda>64thena=a
+32                                       <hb>
150 ifa=20 thena=8                        <oe>
160 print#2,chr$(a);                      <dn>
170 get#2,a$:ifa$=""then130               <pj>
175 ifa$=chr$(10)then170                  <lm>
180 a=asc(a$):b=a:ifa>96thenb=a-32        <pi>
185 ifa<91 and a>64 then b=a+128          <bk>
190 ifa=8then b=20                        <op>
200 a=b:print chr$(a);:goto130            <pb>
210 rem -p r o g r a m m e n d e-         <ae>
```

ASCII-STANDARD-CODE

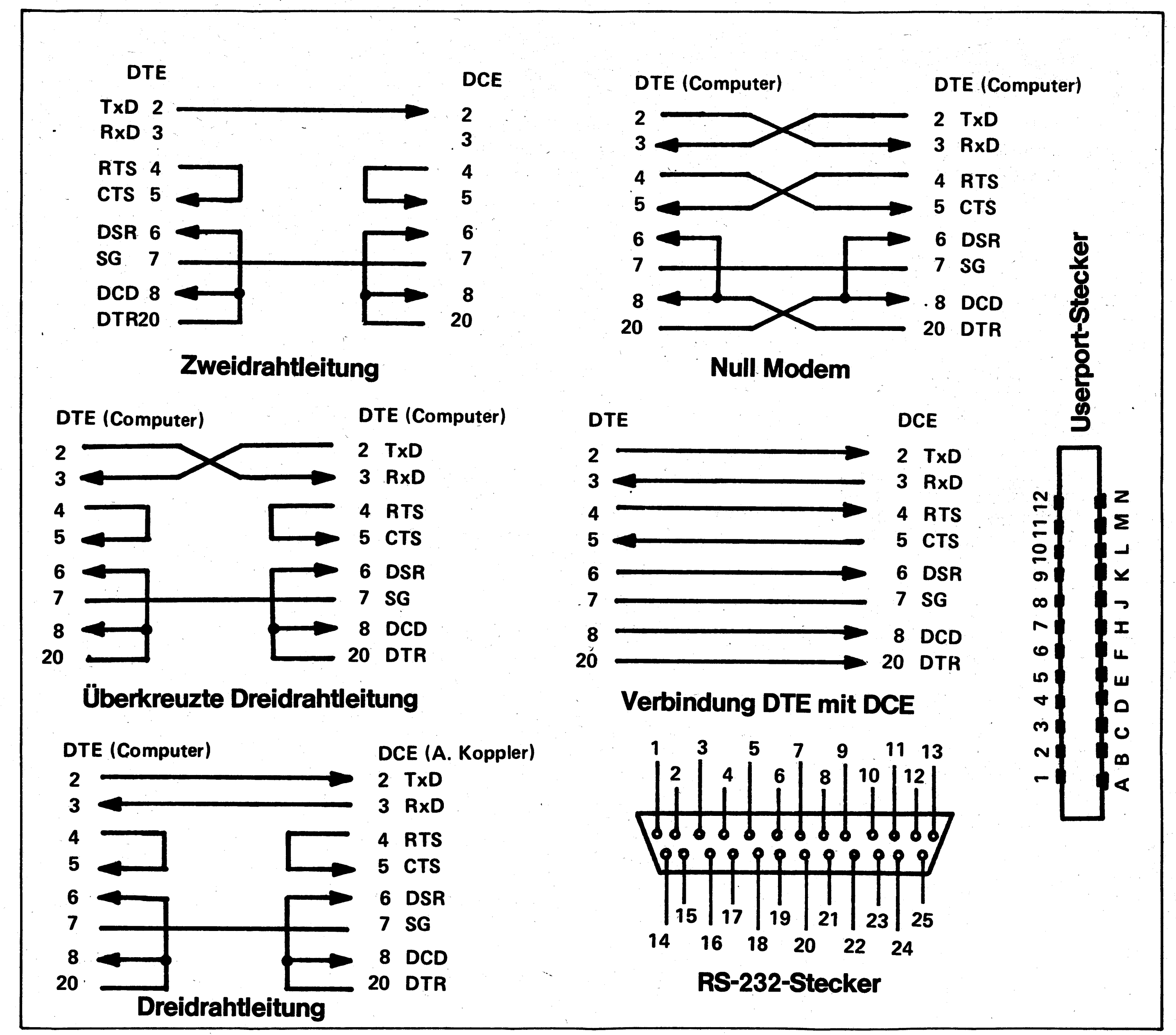
Char Hex Dec Char Hex

Char	Hex	Dec	Char	Hex	Dec
A	41	65	a	61	97
B	42	66	b	62	98
C	43	67	c	63	99
D	44	68	d	64	100
E	45	69	e	65	101
F	46	70	f	66	102
G	47	71	g	67	103
H	48	72	h	68	104
I	49	73	i	69	105
J	4A	74	j	6A	106
K	4B	75	k	6B	107
L	4C	76	l	6C	108
M	4D	77	m	6D	109
N	4E	78	n	6E	110
O	4F	79	o	6F	111
P	50	80	p	70	112
Q	51	81	q	71	113
R	52	82	r	72	114
S	53	83	s	73	115
T	54	84	t	74	116
U	55	85	u	75	117
V	56	86	v	76	118
W	57	87	w	77	119
X	58	88	x	78	120
Y	59	89	y	79	121
Z	5A	90	z	7A	122

Taste“. Daraufhin werden Sie nach dem Filenamen gefragt. Geben Sie den Namen Ihres vorbereiteten Files an und bestätigen Sie mit RETURN.
Ihr File wird nun zur Gegenstelle gesandt.
Unser kleines Terminalprogramm ist natürlich nicht der Weisheit letzter Schluß, es soll Ihnen vielmehr die Grundzüge der Datenfernübertragung demonstrieren und Sie zum weiteren Ausbau des Programmes inspirieren.

AKUSTIKKOPPLER ODER MODEM?

Im Grunde sind auch die Akustikkoppler Modems. Denn das Wort Modem ist die Abkürzung von Modulator-Demodulator. Das, was wir im Hausgebrauch Modem nennen, sind Modems, die galvanisch direkt mit der Telefonleitung verbunden sind.
Im Gegensatz dazu gibt es die Akustikkoppler, die Signale akustisch empfangen und senden.
Bei der Übertragung mit Akustikkopplern gibt es verschiedene Fehlerquellen. Die Probleme beginnen schon bei der Plazierung des Telefonhörers in die Muffen des Akustikkopplers: Meistens ist bei den neuen Apparaten der Post der Hörer zu groß, bei den alten dagegen zu klein. Das hat zur Folge, daß die akustischen Signale, die gesendet und empfangen werden, nicht richtig umgesetzt werden. Die Übertragung ist somit gefährdet, denn schon der kleinste Huster veranlaßt den Akustikkoppler, wirre Zeichen auf den Bildschirm zu bringen und auch zu übertragen.
Abhilfe bringen da die Modems. Da sie direkt an der Telefonleitung angeschlossen sind, fallen die Probleme mit störenden Geräuschen in der Umgebung weg. Gekoppelt mit einem guten Terminalprogramm, können Sie sogar Ihr Modem Telefonnum-



mern wählen lassen und das so lange, bis ein Anschluß erreicht ist. Ihr Computer nimmt Ihnen sogar die lästige „Einlog-Prozedur“ beim Anschluß mit Mailboxen ab. Auch können Sie damit Anrufe von anderen Computerbesitzern empfangen, die eine eigene Mailbox betreiben.

Leider gibt es für die Commodore-Homecomputer wie C128, C64 oder Plus/4 noch keine von der Deutschen Bundespost zugelassenen und mit einer ZZF-Prüfnummer versehenen Modems. Es werden zwar einige von verschiedenen Firmen angeboten, aber diese dürfen Sie nur an Haustelefonanlagen und Nebenstellen betreiben.

Also bleibt nur der gute alte Akustikkoppler und das Warten, bis sich unsere liebe Post endlich entschließt, nicht nur an die professionellen PC-User, sondern auch an die Homecomputer-Besitzer zu denken.

Computer lassen sich aber nicht nur per Akustikkoppler oder Modem miteinander verbinden. Die direkte Verbindung ist immer noch die einfachste Lösung. Diese Art wird auch Null-Modem genannt.

Für die Datenübertragung (Received Data, Transmitted Data, Signal Ground) sieht die RS-232-Norm nicht nur Kabel für die Datenübertragung vor, sondern auch solche für die Steuerung der Kommunikation. Da die Request-to-send (RTS)- und die Celar-to-Send (CTS)-Leitungen nicht eindeutig definiert sind, können die Anschlüsse bei verschiedenen Rechnern unterschiedlich ausfallen.

Verbinden Sie alle Leitungen zwischen den beiden Computern gekreuzt. Also die RTS (request to send) mit der CTS (clear to send)-Leitung und umgekehrt. Die Signal-Erde wird mit der Signal-Erde verbunden. Die Leitungen TXD (Transmitted Data) und RXD (Received Data) werden ebenfalls kreuzweise verbunden.

Funktioniert die Übertragung nicht, so versuchen Sie, die Leitungen ungekreuzt miteinander zu verbinden. Das wäre RTS mit RTS und CTS mit CTS. Sollte sich immer noch kein Erfolg einstellen, so verbinden Sie auch noch die Leitung TXD und RXD direkt. Zum besseren Verständnis betrachten Sie bitte unsere Grafik. □

DFÜ-LEXIKON

Von A bis Z

In vielen Büchern über Datenfernübertragung finden sich unverständliche Fachausdrücke und Abkürzungen. Wir wollen Ihnen einen kleinen Überblick der häufigsten Bezeichnungen geben.

ANSWER
Senden mit dem Antwortfrequenzpaar f1=1850 Hz für 0
f2=1650 Hz für 1.

ASCII
American Standard Code of Information Interchange.
ASCII ist eine genormte Code-Wandlung, die jedem Buchstaben, jeder Zahl und Sonderzeichen einen bestimmten numerischen Wert zuordnet. Dieser Code wird von den meisten Computern verstanden und wird auch als Standard im DFÜ-Bereich verwendet.



BAUD
Gibt die Übertragungsgeschwindigkeit an. Anzahl der Signalwechsel pro Sekunde.

BPS
Eine andere Möglichkeit, die Übertragungsgeschwindigkeit zu bezeichnen.
Anzahl der übertragenen Bits pro Sekunde.

BTX
Bildschirmtext der Post; Übertragung mit 1200/75 Baud.

CARRIER
Carrier bedeutet auf deutsch „Träger“ und bezeichnet im allgemeinen den Ton, mit dem Computer signalisieren, daß sie zur Kommunikation bereit sind. Dieses Carrier-Signal ist ein gleichbleibender Pfeifton mit einer bestimmten Frequenz. Mailboxen zum Beispiel warten bei Anruf, ob ein solcher Carrier vorhanden ist, erst dann beginnen sie mit dem Senden von Daten.

CBBS
Computer Bulletin Board System.

CCITT
Comite Consultative International Telegraphique et Telefonique.
Europäische Norm für Telekommunikation.

CTS
Clear To Send oder empfangsbereit.
Leitung, die anzeigt, daß ein DCE-Gerät bereit ist, Daten vom DTE-Gerät zu empfangen. Diese Leitung ist im Normalfall AUS. Wenn über die RTS-Leitung Sendebereitschaft angezeigt wird, so löst das DCE-Gerät mit CTS EIN das Senden von Daten aus. Ist die RTS-Leitung auf AUS geschaltet, so wird auch die CTS-Leitung auf AUS geschaltet (siehe auch RTS).

CUG
Closed User Group, spezielle Teilnehmergruppe im Datex-P-Netz.

DATEX-P
Datennetz der Deutschen Bundespost.
Übertragungsgeschwindigkeit bis 9600 Baud.

DCD
Empfangspegel vorhanden.
Leitung, über die ein DCE-Gerät einem DTE-Gerät anzeigt, daß es ein Signal empfängt. Dies veranlaßt das DTE-Gerät, ein RTS-Signal zu senden.

DCE
Data Communication Equipment.
Gerät, mit dem Daten empfangen werden können (Modem, Drucker und so weiter).

DSR
Datenempfänger betriebsbereit.
Leitung, die dem DTE-Gerät anzeigt, daß das DCE-Gerät betriebsbereit ist. Im Normalfall ist diese Leitung auf EIN geschaltet.

DTE
Data Terminal Equipment.
Gerät, das Daten sendet, zum Beispiel Terminal, Computer.

DTR
Data Terminal Ready. Das DTE-Gerät zeigt mit einem EIN-Signal an, daß es betriebsbereit ist.

DOWN-LOAD
Bezeichnung für das Einlesen von Daten (Texte, Programme) aus einem anderen Rechner.

FSK
Frequency Shift Keying.
Daten werden durch Frequenzumtastung moduliert.

ISDN
Integrated Services Digital Network.
Sämtliche Daten, auch Sprache, werden digital übertragen.

LOGIN
Verbindungsaufbau zu einem Rechner, meistens mit einer User-ID.

LOGOFF
Gegenteil von LOGIN.
Wird meistens mit den Befehlen LOGOFF oder BYE durchgeführt.

MAILBOXEN
Mailboxen nennt man in unserer heutigen Zeit Computer, die über normalen Telefonanschluß oder DATEX-P angerufen werden können. Mailboxen stellen dem Anrufer

sogenannte „Bretter“ zur Verfügung, auf denen öffentliche Mitteilungen an andere Anrufer geschrieben oder deren Nachrichten abgerufen werden können. Um einen persönlichen „Briefkasten“ in einer Mailbox zu erhalten, ist es in der Regel notwendig, sich als „User“ eintragen zu lassen. In diesen Briefkasten können andere Teilnehmer persönliche Nachrichten schreiben, die nur der eingetragene User lesen kann. Außerdem können Sie Mailboxen als elektronische Tageszeitungen, produziert von den Anrufern, betrachten.

MODEM
Modulator-Demodulator.
1. Das Galvanische Modem:
Es wird direkt an der Telefonleitung angeschlossen. Muß von der Bundespost genehmigt und angeschlossen werden.
2. Der Akustik-Koppler:
Der Telefonhörer wird auf Gummimanschetten des Kopplers aufgesteckt. Keine direkte elektrische Verbindung zum Telefonnetz.
Nur mit ZZF-Nummer der Bundespost genehmigt.

NUA
Network User Adress.
Teilnehmerkennung im DATEX-P-Netz. Gibt die anzurufende Nummer an.

NUI
Network User Identification.
Teilnehmerkennung im DATEX-P-Netz. Gibt die eigene Nummer an. Dient zur Abrechnung der empfangenen Daten durch die Bundespost und kann nur dort beantragt werden.

ORIGINATE
Senden mit dem Sendefrequenzpaar f1=1189 Hz für 0
f2= 980 Hz für 1 (siehe ANSWER).

PAD
Packet Assembly and Dissassembly Facility.
DATEX-P-Vermittlungsstelle.
Übernimmt die Umwandlung des asynchronen Datenflusses in genormte „Pakete“ und umgekehrt.

PARAMETER
Als Parameter werden sämtliche Einstellungen im Übertragungsprotokoll bezeichnet (siehe auch PROTOKOLL).

PARITÄT
1. Gerade Parität:
Das Paritäts-Bit wird gesetzt, wenn die Anzahl der Datenbits im gesendeten Zeichen eine gerade Zahl ergibt.
2. Ungerade Parität:
Ist im gesendeten Zeichen die Anzahl der Datenbits ungerade, wird das Paritätsbit gesetzt.
3. Keine Parität:
Bei der Übertragung wird auf das Senden des Paritätsbits verzichtet.

PROTOKOLL
1. Festlegung der Regeln bei der Datenübertragung (zum Beispiel Baud, Übertragungsart, Parität).
2. Aufzeichnung der empfangenen Daten.

RS-232
Genormte Schnittstelle zur seriellen Übertragung von Daten.

RTS
Ready To Send oder Request To Send (sendebereit).
Leitung, mit der vom DTE-Gerät beim DCE-Gerät angefragt wird, ob Daten gesendet werden können. Im Normalfall ist diese Leitung AUS. Sollen Daten gesendet werden, so wird vom DTE-Gerät ein EIN-Signal an das DCE-Gerät gesendet. Die Übertragung beginnt, wenn das DCE-Gerät über die CTS-Leitung ein EIN-Signal sendet. Beim Beginn der Übertragung wird das RTS-Signal auf AUS gesetzt.

RxD
Recieved Data.
Über diese Leitung werden Daten vom DTE-Gerät zum DCE-Gerät gesendet.

SG
Signal Ground.
Erdungsleitung.

STARTBIT
Zeigt den Beginn eines zu sendenden Zeichens an.

STOPBIT(S)
Kennzeichnung des Übertragungsendes eines Zeichens. Meist werden ein oder zwei Stopbits verwendet.

SysOp
System Operator.
Bezeichnung für den Betreiber einer Mailbox.

TERMINAL
Im allgemeinen wird damit das Programm bezeichnet, das der Computer braucht, um mit anderen Rechnern zu kommunizieren.
Außerdem werden auch die einzelnen Arbeitsplätze einer Großrechenanlage als Terminal bezeichnet.

TxD
Transmitted Data.
Über diese Leitung werden die Daten vom DTE-Gerät zum DCE-Gerät übermittelt.

UP-LOAD
Bezeichnung für das Senden von Daten (Texte, Programme) an einen anderen Rechner.

USER-ID
User Identification, Benutzererkennung.
Nach Eingabe der User-ID erkennt das angerufene System, ob der Anrufer zugriffsberechtigt ist oder nicht.

ÜBERTRAGUNGSARTEN

1. SIMPLEX:
Übertragung nur in eine festgelegte Richtung, vom Sender zum Empfänger, möglich (Rechner–Drucker).

2. HALBDUPLEX:
Datenübertragung in beide Richtungen, aber immer nur abwechselnd. Wird bei sehr hohen Übertragungsgeschwindigkeiten benutzt.

3. VOLLDUPLEX:
Die am meisten benutzte Übertragungsart.
Datenübertragung in beide Richtungen gleichzeitig.

V.21
Protokoll für eine Übertragung mit 300 Baud.

V.24
Europäische Norm der RS-232-Schnittstelle.

X.25
CCITT-Protokoll für synchrone Datenübertragung.

X-MODEM
In vielen neueren Terminalprogrammen für Computer befindet sich der Menüpunkt X-MODEM. Damit lassen sich Programme sofort lauffähig übertragen, ohne den lästigen Umweg über ASCII-Codewandlung.

XON/XOFF PROTOKOLL
Vereinbarte Steuerung des Datenflusses während der Datenübertragung.
XON – (CTRL-Q):
Signal für die Gegenstation zum Senden der Daten.
XOFF – (CTRL-S):
Signal für die Gegenstation, daß sie das Senden von Daten unterbrechen soll.

128 & COM

Jetzt haben auch C-128-Besitzer unter CP/M die Möglichkeit, mit COM Datenfernübertragung zu betreiben. Was auf den ersten Blick recht ansprechend wirkt, denn COM existiert in CP/M-Versionen bereits für fast alle Homecomputer, von Schneider CPCs bis zu MSX.

Vor dem Laden von COM darf man natürlich nicht vergessen, die CP/M-Systemdiskette einzuladen, ansonsten braucht am Anfang nichts beachtet werden. Was bereits zu Anfang ganz besonders auffällt, ist der langsame Bildschirmaufbau. Es soll sich wahrscheinlich um eine neue Art Windowing handeln. Das Hauptmenü umfaßt 6 verschiedene Optionen. Unter den alt bekannten Menüfunktionen taucht da plötzlich eine ganz neue auf, eine mit dem Namen Makros.

WAS SIND MAKROS?

Jetzt stellt sich die Frage: Was sind Makros? Makros sind keine Kekse, sondern kleine Helfer, die das Arbeiten zwischen dem Computer und der Mailbox oder der Datenbank erleichtern. Nehmen wir an, Sie rufen täglich Ihre spezielle Mailbox an, in der Sie immer die gleichen Menüpunkte lesen. Warum sollten dann Sie immer die selben Punkte des Menüs anwählen und danach z.B. auf dem Drucker ausgeben oder der Diskette abspeichern lassen? Diese Arbeit kann jetzt ein Makro übernehmen. Für die Programmierung der Makros gibt es eine eigene Programmiersprache mit 27 Befehlen und Anweisungen. Sogar die Übertragungsparameter lassen sich mit Hilfe eines Befehles einstellen. Bei Übertragungsfehlern kann man die Arbeit mit Makros allerdings vergessen.

MAILBOXDIALOG MIT DRUCKERPROTOKOLL

Mit „2" kommt man in den Terminalmodus, auch von hier aus ist es möglich, bei einem Mailboxdialog den Drucker per Tastendruck zu steuern. Spätestens jetzt bemerkt der Anwender, der mit einem Fernseher arbeitet, daß er einen Monitor braucht, welcher 80 Zeichen darstellen kann. Der Text in den Spalten 41–80 wird zwar übertragen, aber mit 40 Zeichendarstellung kann man ihn nur ungenügend lesen. Hier hilft dann nur das Ausdrucken des gesendeten Textes. Beim Schreiben in eine Mailbox ist die Tastaturabfrage zu schnell, dadurch entstehen manchmal doppelte Zeichen.

WEITERE FUNKTIONEN

Hinter dem Menüpunkt „Dateien" verstecken sich weitere 8 Funktionen, die das Arbeiten mit der Floppy ermöglichen. Es besteht auch die Möglichkeit, das Laufwerk zu wechseln. Es geht aber meiner Meinung schneller, wenn man statt dessen die Disketten wechselt.
Mit „Initialisierung" wird ein Submenü aufgebaut, in dem es möglich ist, die Übertragungsparameter einzustellen. Da gibt es eine Funktion namens Filter, die das Umschalten zwischen ASCII, DIN und ungefiltert erlaubt. Dazu gibt es einen schönen Satz, der mit einer Art Kurzbeschreibung auf der Programmdiskette abgespeichert ist. Er lautet wie folgt: „Da der C-128 unter CP/M nicht über den nötigen Zeichensatz verfügt, steht kein DIN-Filter zur Verfügung. Es wird immer im ASCII-Zeichensatz gearbeitet."
D.h.: Es ist nicht möglich, auf einen deutschen Zeichensatz zurückzugreifen, der im Menü angepriesen wird!
Unter Baudrate kann bis zu 9600 Baud eingestellt werden. Falls der C-128 wirklich so hoch fahren könnte, fragt sich, was das Modem dazu kosten würde. Es ist wahrscheinlicher, mit dieser Baudrate eine reine Rechner-Rechner-Verbindung herzustellen, um schnellen Datentransfer zu ermöglichen.
Doch auch hierfür werden dann bei so hoher Geschwindigkeit Spezialkabel nötig, um die Übertragungssicherheit zu gewährleisten.

STEUERTASTEN WAHLWEISE SCHALTEN

Weiter besteht die Möglichkeit, die Steuertasten ein- und wieder auszuschalten. Bei eingeschalteten Steuertasten können folgende Steuercodes, die eine Mailbox evtl. braucht, nicht gesendet werden: CTRL D, F, G, H, I, J, L, M, Q, S und CTRL T. Unter anderem wird auf der Programmdiskette ein Kopierprogramm mitgeliefert. Es ist allerdings das langsamste Kopierprogramm, welches ich je erlebt habe. Man sollte die Arbeitskopie doch lieber im 128-Modus kopieren, das geht schneller. Wie alle CP/M-Programme wird auch COM ohne Kopierschutz geliefert, was natürlich sehr anwenderfreundlich sein soll. Weniger anwenderfreundlich wird es aber, wenn keine Diskette im Laufwerk liegt und man die Protokolldatei abspeichern will. Das führt dann unwiderruflich zum Ende von COM, die Daten sind natürlich verloren.

KEIN INTEGRIERTER TEXTEDITOR

Es wird aufgefallen sein, daß es keinen Menüpunkt gibt, in dem es möglich ist, einen Text einzugeben, zu ändern oder einen gerade empfangenen Text zu editieren.
Dies ist richtig, es wird allerdings ein Textprogramm mitgeliefert, das funktionsmäßig an WordStar angepaßt ist.
Das Handbuch: Eines der besten Handbücher, die ich je gelesen habe. Das Buch ist in mehrere übersichtliche Kapitel unterteilt. Im ersten Kapitel finden DFÜ-Neulinge die nötige Rohstoffinformation die man braucht, um mit Mailboxen und Datenbanken effektiv arbeiten zu können. Auch was es mit DATEX-P auf sich hat, wurde ausführlich abgedruckt. Selbst die Übertragung zwischen Userport und Mailbox wird sozusagen Bit für Bit erklärt. Im dritten Kapitel steht ein Mastermailboxdialog. Darüber wird sich der Anfänger sehr freuen, der noch keinen Kontakt mit einer Mailbox hatte. Da wird z.B. auch die Programmierung eines Makros Punkt für Punkt erarbeitet. Am Ende des Handbuches befinden sich mehrere Anhänge, wie z.B. Pannenhilfe und ein Mailboxführer.
COM ist ein Terminalprogramm unter CP/M, das meiner Meinung nach noch ganz am Anfang steht. Der wesentliche Nachteil besteht in der Geschwindigkeit. Das gilt nicht nur für die Menüfenster, sondern auch für die zwei mitgelieferten Programme COPYDISK und EDIT.
Es beinhaltet zwar sämtliche Funktionen, die zu einem erfolgreichen Mailboxdialog notwendig sind (außer ein eingebauter Texteditor), ist aber teilweise umständlich zu bedienen und recht langsam im Aufbau der Menüfunktionen. Lediglich das hervorragende Handbuch verdient uneingeschränktes Lob.

Andree Hermann

BASIC TOKEN V7,0

Alle Commodore Basics inklusive der Original-Version des Old Rom der Pet Serie 2001 benutzen in ihrem Basic bestimmte Schlüsselwörter (Kommandos, Statements und Funktionen).
Diese Schlüsselwörter, auch TOKEN genannt, werden vom Computer benutzt, um die entsprechenden Basicbefehle platzsparend im Speicher abzulegen. Ein Token besteht aus einem oder höchstens zwei Byte, die durch diese Codierung natürlich sehr viel Platz sparen. Nimmt man z.B. den Basicbefehl 'PRINT', der eigentlich aus 5 Byte besteht, aber vom Computer nur als ein Byte abgelegt wird. Gleichgültig, ob der Befehl ausgeschrieben wird oder ob man die Abkürzung '?' benutzt. Alle diese Basicbefehle sind in einer Token-Tabelle abgelegt und haben immer einen Codierungswert größer 128 (dez) bzw. 80 (hex). Mit der Kenntnis dieser Werte ist es sogar möglich, ein Programm zu ändern. Man kann z.B. ein 'PRINT' durch ein 'REM' ersetzen oder ein 'RETURN' durch ein 'LIST' usw. Um dieses zu

SO WERDEN BEFEHLE ABGELEGT

erreichen, muß man natürlich wissen, wie der Computer die Befehle im Basicspeicher ablegt. Dazu tippen Sie einmal folgende Basiczeile ein.

10 PRINT"HALLO"
>RETURN<

Nun wollen wir uns einmal anschauen, wie der Computer diese Zeile

Commodore Basic V7.Ø Token

HAUPTSÄCHLICHE SCHLÜSSELWÖRTER

Token	Befehl	Token	Befehl	Token	Befehl
8Ø	END	D6	RESUME	FE Ø2	BANK
81	FOR	D7	TRAP	FE Ø3	FILTER
82	NEXT	D8	TRON	FE Ø4	PLAY
83	DATA	D9	TROFF	FE Ø5	TEMPO
84	INPUT#	DA	SOUND	FE Ø6	MOVSPR
85	INPUT	DB	VOL	FE Ø7	SPRITE
86	DIM	DC	AUTO	FE Ø8	SPRCOLOR
87	READ	DD	PUDEF	FE Ø9	RREG
88	LET	DE	GRAPHIC	FE ØA	ENVELOPE
89	GOTO	DF	PAINT	FE ØB	SLEEP
8A	RUN	EØ	CHAR	FE ØC	CATALOG
8B	IF	E1	BOX	FE ØD	DOPEN
8C	RESTORE	E2	CIRCLE	FE ØE	APPEND
8D	GOSUB	E3	GSHAPE	FE ØF	DCLOSE
8E	RETURN	E4	SSHAPE	FE 1Ø	BSAVE
8F	REM	E5	DRAW	FE 11	BLOAD
9Ø	STOP	E6	LOCATE	FE 12	RECORD
91	ON	E7	COLOR	FE 13	CONCAT
92	WAIT	E8	SCNCLR	FE 14	DVERIFY
93	LOAD	E9	SCALE	FE 15	DCLEAR
94	SAVE	EA	HELP	FE 16	SPRSAV
95	VERIFY	EB	DO	FE 17	COLLISION
96	DEF	EC	LOOP	FE 18	BEGIN
97	POKE	ED	EXIT	FE 19	BEND
98	PRINT#	EE	DIRECTORY	FE 1A	WINDOW
99	PRINT	EF	DSAVE	FE 1B	BOOT
9A	CONT	FØ	DLOAD	FE 1C	WIDTH
9B	LIST	F1	HEADER	FE 1D	SPRDEF
9C	CLR	F2	SCRATCH	FE 1E	QUIT
9D	CMD	F3	COLLECT	FE 1F	STASH
9E	SYS	F4	COPY	FE 2Ø	-
9F	OPEN	F5	RENAME	FE 21	FETCH
AØ	CLOSE	F6	BACKUP	FE 22	-
A1	GET	F7	DELETE	FE 23	SWAP
A2	NEW	F8	RENUMBER	FE 24	OFF
CB	GO	F9	KEY	FE 25	FAST
D5	ELSE	FA	MONITOR	FE 26	SLOW

SEKUNDÄRE SCHLÜSSELWÖRTER

Token	Befehl	Token	Befehl	Token	Befehl
A3	TAB(	A7	THEN	FC	UNTIL
A4	TO	A8	NOT	FD	WHILE
A5	FN	A9	STEP		
A6	SPC(	AA	USING		

ARITHMETISCHE SCHLÜSSELWÖRTER

Token	Befehl	Token	Befehl	Token	Befehl
AA	+	AC	*	AE	↑
AB	-	AD	/		

LOGISCHE SCHLÜSSELWÖRTER

Token	Befehl	Token	Befehl	Token	Befehl
AF	AND	BØ	OR		

FUNKTIONS SCHLÜSSELWÖRTER

Token	Befehl	Token	Befehl	Token	Befehl
B4	SGN	C2	PEEK	D2	HEX$
B5	INT	C3	LEN	D3	ERR$
B6	ABS	C4	STR$	D4	INSTR
B7	USR	C5	VAL	CE Ø2	POT
B8	FRE	C6	ASC	CE Ø3	BUMP
B9	POS	C7	CHR$	CE Ø4	PEN
BA	SQR	C8	LEFT$	CE Ø5	RSPPOS
BB	RND	C9	RIGHT$	CE Ø6	RSPRITE
BC	LOG	CA	MID$	CE Ø7	RSPCOLOR
BD	EXP	CC	RGR	CE Ø8	XOR
BE	COS	CD	RCLR	CE Ø9	RWINDOW
BF	SIN	CF	JOY	CE ØA	POINTER
CØ	TAN	DØ	RDOT		
C1	ATN	D1	DEC		

VERGLEICH SCHLÜSSELWÖRTER

Token	Befehl	Token	Befehl	Token	Befehl
B1	>	B2	=	B3	<

Commodore Basic V7.Ø Token

Monitorauszug:

```
>Ø1CØØ ØØ ØE 1C ØA ØØ 99 22 48 : ......"H
>Ø1CØ8 41 4C 4C 4F 22 ØØ ØØ ØØ : ALLO"...
```

So sieht der Monitorauszug für unsere Basiczeile aus.
Aufgeteilt ergibt sich dann folgendes Bild:

```
>Ø1CØØ ØØ   :Zeichen für Basiczeile
>Ø1CØ1 ØE   :Linkadresse auf nächste Zeile
>Ø1CØ2 1C   :Linkadresse auf nächste Zeile
>Ø1CØ3 ØA   :Zeilennummer (1Ø)
>Ø1CØ4 ØØ   :Leerstelle
>Ø1CØ5 99   :Token für PRINT
>Ø1CØ6 22   :"
>Ø1CØ7 48   :H
>Ø1CØ8 41   :A
>Ø1CØ9 4C   :L
>Ø1CØA 4C   :L
>Ø1CØB 4F   :O
>Ø1CØC 22   :"
>Ø1CØD ØØ   :Zeilenende
>Ø1CØE ØØ   :Neue Zeile
>Ø1CØF ØØ   :Programmende(letzte Zeile)
```

Anhand dieser Tabelle kann man nun genau sehen, wie der Computer Basiczeilen im Speicher ablegt.

im Speicher abgelegt hat. Dazu eignet sich der eingebaute Maschinensprachemonitor natürlich hervorragend. Mit der Funktionstaste 'F8' wird er gestartet. Nachdem das Einschaltbild des Monitors erschienen ist, geben Sie dann folgenden Befehl ein.

M 1C00 >RETURN<

Es erscheint nun ein Ausschnitt aus dem Speicher mit Adressennummer und den entsprechenden Werten in hexadezimaler Schreibweise. Fangen wir am besten von vorne an. In der ersten Speicherstelle ist eine '0' abgelegt. Dies sagt dem Microprozessor, daß an dieser Stelle eine Basiczeile beginnt. In den nächsten beiden Speicherstellen wird angezeigt, wo die nächste Basiczeile beginnt. Diese Adresse nennt man auch LINK-Adresse. Diese Adresse ist übrigens beim LISTEN eines Programmes ganz wichtig! Durch Verstellen dieser Zeiger innerhalb eines Programmes können ganze Teile vor dem Listen geschützt werden! In den nächsten Speicherstellen finden wir die Zeilennummer unserer Basiczeile wieder.
In der nächsten Stelle steht dann unser 'TOKEN' als Hexzahl. Nämlich die '83', die als Codierung für 'PRINT' steht. Daran anschließend folgen die Werte für das Wort 'HALLO' und die Anführungszeichen. Am Ende der Programmzeile stehen dann eine oder mehrere Nullen. Eine Null kennzeichnet das Ende der Programmzeile.

TOKENÄNDERUNG IST EINFACH

Ist diese Null die letzte eines Programmes überhaupt, so kommen noch zwei weitere Nullen hinzu. Aber zurück zu unseren TOKEN.
Die für uns wichtige Speicherstelle heißt '1C05'. Hier stehen das TOKEN für PRINT. Wird nun in diese Speicherstelle ein anderer Wert geschrieben, z.B. (83), wird automatisch auch die Programmzeile verändert. Geben Sie nun mal statt der '99' eine '83' ein, RETURN nicht vergessen, und verlassen dann den Monitor mit X >Return<. Nun listen Sie einmal die anfangs eingegebene Basiczeile und schauen sich an, was sich da getan hat. Und o Wunder, jetzt steht da plötzlich statt 'PRINT' ein 'DATA' vor dem Wort HALLO. Verblüffend nicht wahr? Versuchen Sie ruhig noch ein paar andere Möglichkeiten. Wenn Sie das Basicprogramm starten, kann es natürlich zu der einen oder anderen Fehlermeldung kommen. Diese Änderungen können natürlich auch aus einem Programm heraus erfolgen. Große Programmgeneratoren, welche sich selbst erweitern oder verändern, benutzen diese Möglichkeiten. Hier sind noch viele Anwendungen denkbar.

TIPS, TRICKS, PEEKS UND POKES FÜR DEN 128

Ein sehr komfortabler Fortschritt des C-128 ist die Möglichkeit, die Funktionstasten F1 – F8 bereits mit Text belegt vorzufinden. Sollte Ihnen der nicht passen (jede Programmanwendung benötigt andere, spezifische Tastendrucke), so können Sie diese jederzeit ändern. Rufen Sie diese Belegung durch die Anweisung "KEY" auf, der Text dieser acht Tasten erscheint auf Ihrem Bildschirm, und läßt sich nun von Ihnen umschreiben. Es ist z.B. ohne weiteres möglich, einen Text von bis zu 256 Zeichen auf eine Funktionstaste zulegen, aber: insgesamt darf die Zahl der Zeichen oder Befehle, die Sie auf die F-Tasten legen, auch nicht größer sein als 256 Bytes. Der entsprechende Speicherbereich für diese „Tabelle" liegt bei $1000 (4096) – $10FF (4351).
Den neuen, geänderten Funktionstastentext speichern Sie bitte mit folgendem Befehl ab:
BSAVE"F-TASTEN", ON B0, P 4096 TO P 4351
(Sie können natürlich jeden anderen File-Namen wählen.) Immer, wenn Sie diese spezifizierte Funktionstasten-Belegung in Ihrem C 128 brauchen, laden Sie vorher dieses File wieder mit:
BLOAD"F-TASTEN"

1328 BLOCKS AUF DISKETTE – AUCH IM C-64-MODUS

Wer mit der Diskettenstation 1571 arbeitet, muß auch im C-64-Modus nicht auf die beidseitige Nutzung der eingelegten 5.25-Zoll-Diskette verzichten, er kann die Vorteile der beiden Schreib-/Leseköpfe dieser Floppy ausnutzen. Normalerweise besteht nämlich folgender Unterschied: Formatieren Sie eine Diskette im 128er-Modus, so erhalten Sie nach anschließendem Anzeigen des „Directory" die Meldung: 1328 Blocks frei, nach dem Formatieren im C-64-Modus allerdings nur 664 Blocks. Das liegt daran, daß hier nur eine Seite (die Discunterseite) mit Spuren und Sektoren beschrieben wurde, die Oberseite aber nicht. Überlisten Sie den C 64 doch mit folgender Befehlseingabe:
OPEN 15,8,15,"U0 > M1": CLOSE 15
Nun können Sie auch als C-64-Nutzer beide Diskettenseiten „in einem Durchgang" beschreiben, ohne daß Sie sie umdrehen oder eine zusätzliche Schreibkerbe anbringen müssen.
Wohlgemerkt: es funktioniert nur bei der Floppy 1571, nicht aber bei den Modellen 1570 oder 1541!

UMSCHALTUNG ZWISCHEN GROSS- UND KLEINSCHRIFT

Dies bewirkt beim C 128 der einfache Befehl: PRINTCHR$ (14) oder die manuelle Umschaltung der beiden Tasten „Commodore-Shift". Sollen nun diese Tasten gesperrt bleiben, schreiben Sie anschließend noch: PRINT CHR$(11), diese Eingabe blockiert die Umschaltmöglichkeit dieser beiden Tasten, aufgehoben wird dieser Zustand wieder mit "PRINT CHR$(12)". Da aber viele „Wege zur Zentralrecheneinheit" Ihres C 128 möglich sind, können Sie auch andere Eingaben (z.B. in eigenen Programmen benutzen):

SYS DEC("C880")
Umschalten auf Kleinschrift-Modus

SYS DEC("C892")
Umschalten auf Großschrift

SYS DEC("C8AC")
Commodore-Shift-Taste sperren

POKE 247,128
dto.

SYS DEC("C8A6")
Commodore-Shift-Taste entriegeln

POKE 247,0
dto.

WEITERE TASTATUR-SYSTEMADRESSEN

Nur im 80-Zeichenmodus des C 128 lassen sich folgende Funktionen aufrufen:

SYS DEC("C8C7")
schaltet „Unterstreichen" von Text ein

SYS DEC("C8CE")
hebt diese Option wieder auf

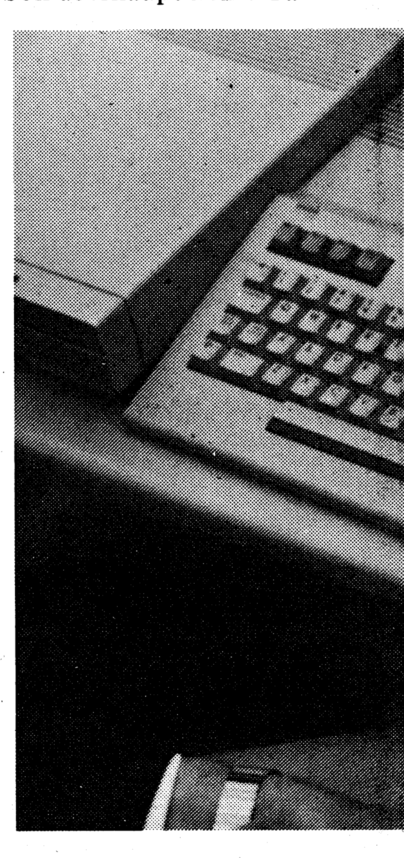
SYS DEC("C8D5")
„Blink"-Modus ein

SYS DEC("C8DC")
„Blink"-Modus aus

Viele Tastaturmöglichkeiten werden mit der sog. ESC-Funktion (CHR$ (27)) in Verbindung mit einem entsprechenden Buchstaben aufgerufen (siehe Handbuch zum C 128, Kap. 4–4 ff).
Die meisten von ihnen können auch im 40-Zeichenmodus genutzt werden, z.B. PRINTCHR$ (27) + "E" setzt den Cursor-Modus von Blinken auf Konstantanzeige. Auch hier gibt es im Betriebssystem des Bildschirmeditors eine SYS-Adresse ($ C01E oder 49182), die dies ebenfalls bewerkstelligt. Sie müssen lediglich den ASCII-Wert des folgenden Buchstabens als Parameter übergeben. Unser genanntes Beispiel würde dann so lauten:

SYS DEC("C01E"),45

Zudem gibt es eine Speicherstelle beim C 128, die die Tastenwiederholfunktion kontrolliert (2594). Ihr Normalwert ist immer "128", es ist also nur das 7. Bit eingeschaltet. Alle Tasten auf der Normaltastatur haben somit eine Wiederholfunktion, wenn Sie diese Taste länger gedrückt halten. Löschen Sie dieses Bit, also "POKE 2594,0", so werden nur die Cursor-, die DEL-, INST- und die SPACE-Tasten wiederholt, so wie's im Prinzip der C 64 kann. Soll überhaupt keine Ta-



Der beste Homecomputer, den Commodore je gebaut hat: der 128 PC. Er stellt gleich drei verschiedene Systeme in einem Gehäuse dar.

ste wiederholt werden, so dürfen Sie lediglich das 6. Bit einschalten, also "POKE 2594,64".
Im 80-Zeichenmodus können Sie auch Ihren Cursor wie bei einem professionellen PC aussehen lassen. – in Strichdarstellung. Der Befehl dazu lautet:
POKE 2603,71.

LISTSCHUTZ BEI BASIC-PROGRAMMEN

Schon seit Zeiten des C 64 waren viele „Freaks" bemüht, neugierigen Zeitgenossen Einblick in ihre Programmierarbeit zu verwehren.
LIST- (und KOPIER-Schutz) haben eine Perfektion erreicht, die nur noch von ausgefuchsteren Usern „geknackt" und überlistet wird. Nun, darum wollen wir uns hier nicht kümmern, sondern Ihnen lediglich einige Anregungen bieten, wie Sie unter Verwendung von System-Adressen und POKEs auch einen leidlichen LIST-Schutz erreichen können. Wichtig ist hierbei, daß es sich durchweg um Befehle handelt, die in den ersten Zeilen Ihres Programms untergebracht werden sollten, um nach dem Start mit "RUN" sofort Ihre Wirkung zu bekommen. Falls das Programm nach dem Laden nicht sofort gestartet wird (auch mit einem „Autostarter" möglich), ist es natürlich jederzeit list- und veränderbar!
Der Inhalt von vier Adressen in der erweiterten Zeropage des C 128 kann für unsere „Manipulationen" verwendet werden:

774 Low-Byte LIST-Routine
775 High-Byte LIST-Routine



808 Low-Byte Interrupt-Routine
809 High-Byte Interrupt-Routine

Diese Adressen haben im Low- und High-Byte-Format die Einsprungstelle im Betriebssystem abgespeichert, die der C 128 aufrufen muß, um die LIST- oder die Interrupt-(Unterbrechungs-)routine im ROM abarbeiten zu können. Schreiben Sie in diese „Merk"-Stellen einen anderen Wert (meist genügt schon die Speicherstelle für das Low-Byte), werden auch andere Systemroutinen aufgerufen, die etwas ganz anderes bewirken.

POKE 774,0: POKE 775,224
(bei Eingabe von "List" wird ein Reset, ein System-Neustart durchgeführt, das Programm, das Sie „listen" wollten, ist gelöscht)

POKE 774,59
(nach "List" erscheinen nur die Zeilennummern, aber kein Listing-Text)

POKE 774,64
(nur die erste Zeilennummer wird gelistet, das Programm hält an und der Cursor verschwindet. Dieser Zustand kann nur mit der Tastenkombination STOP-RESTORE wieder aufgehoben werden)

POKE 774,156
(die Zeilennummer werden normal gelistet, der Programmtext ist jedoch nicht zu gebrauchen!)

POKE 774,81
(jeder vorher verwandte Listschutz-Poke wird damit hinfällig, weil dies der Normalinhalt dieser Speicherstelle ist. Es kann wieder normal gelistet werden)

Die wohl wirksamste Methode, ein Programm (allerdings muß es auch hier erst gestartet werden) zu schützen, ist das „Lahmlegen" der Tastenkombination "STOP/RESTORE", deren Betätigung den NMI- (Nicht maskierten)-Interrupt auslöst, mit dem Sie praktisch aus jedem laufenden Programm herauskommen sollten, egal, was es gerade macht.
Hier genügt es, einfach die Speicherzelle 808 zu „manipulieren".

POKE 808,100
sperrt die STOP/RESTORE-Taste.

Da jetzt ein laufendes Programm nicht mehr angehalten werden kann, Sie also nicht mehr in den direkten Eingabemodus kommen können (mit blinkendem Cursor und der "READY."-Anzeige), bleibt nur noch, den Computer auszuschalten. Das Programm aber, das nicht gelistet werden sollte, ist damit auch verschwunden. Doch schließlich ist das ja der Sinn der Übung, stimmt's? Übrigens, mit "POKE 808,110" stellen Sie den Normalzustand dieser Speicherstelle wieder her, aber den hätten Sie nach erneutem Einschalten des C 128 sowieso.

BASIC-ZEILEN VERSTECKEN

In Verbindung mit den vorher genannten „Programm-Non-Stop"-POKEs, bei denen es darauf ankam, diese in einer der ersten Zeilen Ihres zu schützenden Basic-Programms zu verstecken, stört natürlich ein bißchen die Tatsache, daß jeder „Freak" das Programm laden und per "LIST"-Befehl betrachten kann. Kennt er nun auch diese bestimmten POKEs, ist es ein leichtes für ihn, diese Zeilen zu löschen und das Programm wieder „jungfräulich" zu gestalten. Es bleibt uns also gar nichts anderes übrig, als diese bestimmten Basic-Zeilen zu verstecken. (Das funktioniert in der folgend beschriebenen Weise nicht nur im C 128, sondern auch im C 64 genauso!) Rechts oben auf Ihrer Tastatur finden Sie eine Taste mit der Aufschrift "INST/DEL". Drücken Sie dazu die Shift-Taste, da können Sie innerhalb einer Basic-Zeile Leerzeichen einfügen, ohne „Shift" passiert das Gegenteil, d.h., bestehende Zeichen werden gelöscht. Gehen Sie nun beim Generieren der „Schutzzeile" wie folgt vor:

1) Wählen Sie eine Zeilennummer, die zwischen einem gewissen, von Ihnen gewählten Zählalgorithmus liegt, z.B. bei einer Zehner-Zeilennumerierung Zeilennummer "11" oder "12". Tippen Sie diese Zahl ein. Schreiben Sie dahinter Ihren POKE-Befehl, also 11 POKE 808,100 Drücken Sie jetzt die Shift-Taste und ca. 30-mal die Taste INST/DEL. Es tut sich nichts auf Ihrem Bildschirm, stimmt's, der Cursor bleibt trotz Ihres eifrigen Tastendrückens immer an derselben Stelle, doch das scheint nur so! Tippen Sie nun weiter einen Doppelpunkt, ein "REM" und dahinter das Anführungszeichen oben. Jetzt kommt's: tippen Sie nun ständig die Taste "DEL" (ohne „Shift") und Sie stellen fest, daß nun eine ganze Reihe inverse "T" erscheint, sooft, wie Sie vorher die SHIFT/INST-Tastenkombination betätigt haben, abzüglich des Platzes für den Doppelpunkt, das REM und die Anführungszeichen. Damit haben Sie den Computer die Anweisung gegeben, genauso oft ein Zeichen in dieser Listingzeile zurückzugehen, wie Sie reverse "T" dort stehen haben. Die Zeile würde dann etwa so aussehen:

```
11 POKE 808,100:
   REM"TTTTTTTTTT
   TTTTTTTTTTTTTT
   TTTTT
```

Das letzte Anführungszeichen hinter den "T"s können Sie weglassen! Sie haben damit einen DEL-String geschaffen, den der Computer aber nicht beim Programmablauf, sondern bereits bei Eingabe des Befehls "LIST" ausführt! Als Maßgabe können Sie davon ausgehen, daß etwa doppelt soviel reverse "T" hinter dem auszuführenden Zeilentext stehen sollten (in diesem Falle eben "POKE 808, 100"), um die Zeile inklusive Zeilennummer vollständig verschwinden zu lassen.

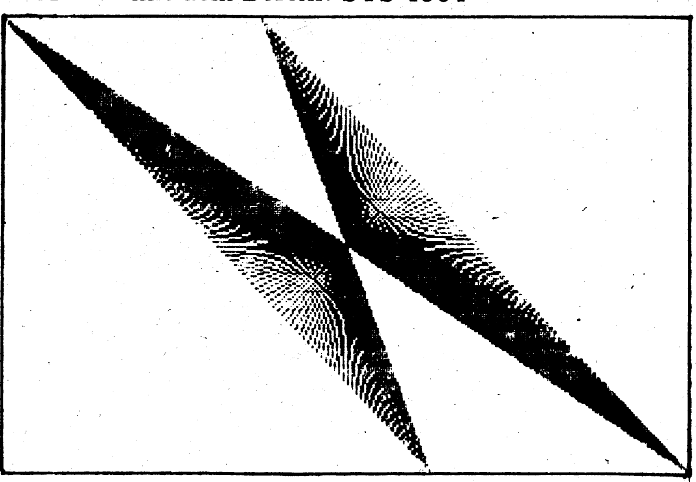
HI-LOW PRINT

Grafiken und Bilder lassen sich mit dem C 128 leicht mit den Befehlen des Basic 7.0 erstellen, ebenso leicht auf Disk speichern und wieder laden, schwieriger wird's allerdings, dieses „Kunstwerk" auf einem Drucker auszugeben. Im Prinzip ist dies mit jedem möglich, der die Einzelbitgrafik integriert hat, wie Epson, Seikosha, Panasonic, Star usw., nur die dazugehörigen Befehle mitsamt den Steuerzeichencodes in Basic zu programmieren, ist nicht ganz so einfach, da der Drucker seine Bytes, die er ausdrucken soll, nach anderen Gesichtspunkten errechnet als der Grafikbildschirm. Abgesehen davon, daß eine entsprechende Umrechnungsroutine erst programmiert werden muß, ist diese in Basic so langsam, daß der Bildausdruck schier endlos dauern kann.

Die Druckroutine in Maschinensprache, die hier abgedruckt ist, bringt Ihnen Ihren Grafikbildschirm in einem Bruchteil der Zeit auf den Drucker, der allerdings in diesem speziellen Fall einer von Epson oder ein dazu kompatibler sein muß. Nach dem Laden und Starten des Basicladers wird das Programm initialisiert und belegt nun den Speicher Ihres C 128 von Adresse 4864 bis 5357.

Zwei Druckmöglichkeiten gibt es:

1) Ausdruck des hochauflösenden Grafikbildschirms (egal, ob HIRES oder Multicolorgrafik)
   Aufruf mit dem Befehl: SYS 4864

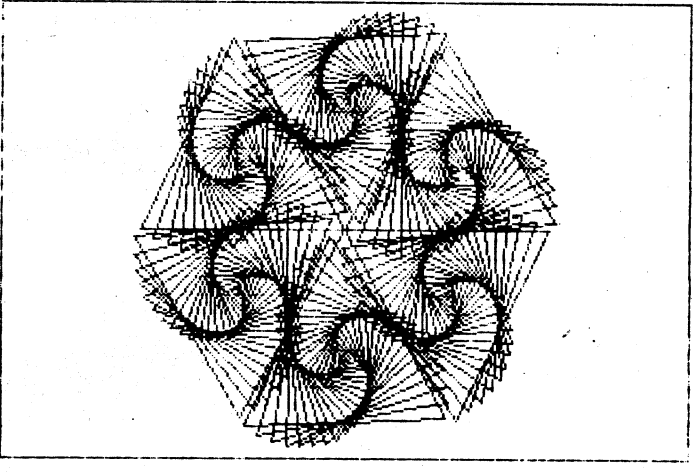


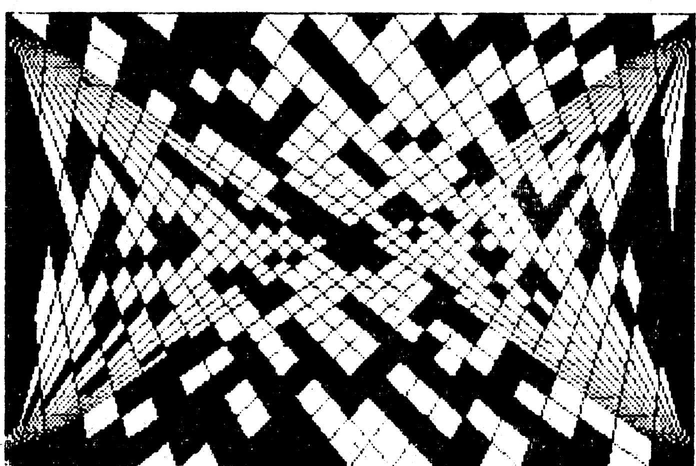
2) der aktuelle Textbildschirm wird ausgedruckt. Der ist nach Einschalten des Computers aktiv oder nach dem Befehl GRAPHICO, wenn Sie aus dem hochauflösenden Modus wieder zurückgekehrt sind.
   Aufruf mit: SYS 5254

Für welche Ausdruckmöglichkeit Sie sich auch entscheiden, die beiden genannten SYS-Befehle, im Direktmodus oder per Programm eingegeben, machen's möglich. Zum Unterprogramm für den Ausdruck des Textbildschirms ist noch zu erwähnen, daß hier grundsätzlich der Groß-/Blockschriftmodus eingestellt ist. Zuständig dafür ist die Speicherstelle 5259, hier steht eine „0". Das bedeutet nichts anderes als die Sekundäradresse „Null" beim Öffnen des Druckerkanals. Falls Sie z.B. für die Klein-/Großschrift eine andere Sekundäradresse benötigen, etwa „7", so POKEn Sie bitte diese Zahl nach der Initialisierung von „Hi-Low-Print" in die Speicherstelle 5259. (POKE 5259,7). *(hb)*

HI-LOW-Print 128 — 1 —

```
100 rem high resolution/        <hk>
110 rem low resolution          <ej>
120 rem hardcopy 128 pc         <kp>
121 rem nur 40-zeichenmodus!    <mc>
130 rem (c) commodore welt/hb   <fi>
140 rem ===================== <bn>
150 rem hir.hardcopy=sys 4864   <bg>
160 rem lor.hardcopy=sys 5254   <ag>
170 rem ===================== <mi>
180 rem druck-sekundaeradresse  <gj>
190 rem fuer lores-hardcopy     <ml>
191 rem auf grosschrift         <dc>
192 rem eingestellt ! (=0)      <kj>
200 rem aendern:$ 148b (5259)   <gm>
210 rem (5.data in zeile 410)   <cd>
211 rem direkt aendern im       <he>
212 rem listing oder:           <ba>
213 rem nach initialisierung    <df>
214 rem mit 'run'               <mn>
215 rem 'poke 5259,zahl'        <gc>
220 rem ===================== <jo>
230 fori=4864to5357:readd$:pok
ei,dec(d$):next:new             <ij>
240 data85,5b,ad,69,14,f0,22,7
8,a9,ff,8d,03,dd,ad,02,dd,09,0
4,8d,02,dd,ad,00                <bo>
250 datadd,09,04,8d,00,dd,a9,1
0,8d,0d,dd,ad,0d,dd,58,4c,44,1
3,20,cc,ff,a9,04                <jk>
260 dataae,6a,14,ac,6b,14,20,b
a,ff,a9,00,20,bd,ff,20,c0,ff,a
2,04,20,c9,ff,a9                <lb>
270 data04,85,5e,a0,03,20,22,1
4,a9,80,85,5f,a9,02,85,60,a5,5
b,29,c0,d0,04,46                <in>
280 data5f,46,60,a5,5b,29,07,a
2,00,20,30,14,a5,5b,30,03,4a,4
a,4a,29,07,a2,02                <em>
290 data20,30,14,a9,19,85,5c,a
5,5f,30,0a,a0,14,a9,20,20,47,1
4,88,d0,fa,a0,08                <jn>
300 data20,22,14,a5,5f,20,47,1
4,a5,60,20,47,14,a9,28,85,5d,7
8,a9,34,85,01,a0                <bj>
310 data07,b1,57,99,7d,14,88,1
0,f8,a9,37,85,01,58,a5,57,18,6
9,08,85,57,90,02                <md>
320 datae6,58,a0,08,a2,00,3e,7
d,14,08,2a,e8,28,24,5b,30,06,e
0,08,d0,f1,f0,08                <nf>
330 data2a,e4,5e,d0,ea,20,47,1
4,20,47,14,88,d0,df,c6,5d,d0,b
a,a5,5f,10,26,a2                <jf>
340 data01,b4,57,b5,59,95,57,9
4,59,ca,10,f5,a5,5e,49,0c,85,5
e,c9,04,f0,0f,a5                <fk>
350 data5b,30,03,4c,97,13,a0,0
0,20,22,14,4c,7a,13,c6,5c,d0,f
4,a0,00,20,22,14                <om>
360 dataa0,0d,20,22,14,20,cc,f
f,a9,04,20,c3,ff,60,b9,6c,14,c
9,ff,f0,06,20,47                <fl>
370 data14,c8,d0,f3,60,a0,06,d
9,96,1f,f0,03,88,d0,f8,a9,00,9
5,57,18,69,20,88                <ak>
380 data10,fb,95,58,60,ae,69,1
4,d0,03,4c,d2,ff,48,8d,01,dd,a
d,00,dd,29,fb,8d                <ck>
390 data00,dd,09,04,8d,00,dd,a
d,0d,dd,29,10,f0,f9,68,60,00,0
4,01,0d,0a,ff,1b                <bb>
400 data33,17,ff,ff,1b,2a,04,f
f,ff,1b,32,ff,ff,00,00,00,00,0
0,00,00,00,00,a9                <oj>
410 data7f,a2,04,a0,00,20,ba,f
f,a9,00,20,bd,ff,20,c0,ff,b0,4
a,a9,00,85,fd,a9                <me>
420 data04,85,fe,a2,7f,20,c9,f
f,a2,19,a9,0d,20,d2,ff,20,e1,f
f,f0,31,a0,00,b1                <fa>
430 datafd,85,fc,29,3f,06,fc,2
4,fc,10,02,09,80,70,02,09,40,2
0,d2,ff,c8,c0,28                <oh>
440 datad0,e6,98,18,65,fd,85,f
d,90,02,e6,fe,ca,d0,cd,a9,0d,2
0,d2,ff,20,cc,ff                <oo>
450 dataa9,7f,4c,c3,ff,c9,08,d
0,f2,60,00                      <bj>
```





VDC-Basic

Der C128 ist aufgrund seiner grafischen Fähigkeiten und den recht komfortablen Befehlen dafür sehr beliebt.

Einen Haken hat das Ganze: Grafik können Sie nur in Verbindung mit dem 40-Zeichen-Bildschirm darstellen. Sagen wir besser, konnten, denn mit diesem Programm geht's nun auch im 80-Zeichen-Modus und einer Auflösung von 640 x 200 Bildpunkten.
Tippen Sie dazu bitte das Hauptprogramm ab, starten es und geben dann in diesem Modus das Listing des Demo-Programmes ein. Sie werden staunen.
Beachten Sie aber bitte unbedingt: Das Demo-Programm darf nur im aktiven Modus der BASIC-Erweiterung eingegeben, gespeichert und geladen werden, da sonst verschiedene neue Befehle nicht richtig interpretiert werden.
Wenn Sie sich das Listing dieses Beispielprogramms ein wenig näher betrachten, wird es ein Leichtes für Sie sein, selbst solche Grafik-Demos mit den zusätzlichen Befehlen (es sind genau achtzehn) dieser BASIC-Erweiterung speziell mit dem VDC-Chip zu erstellen.
VDC-BASIC belegt den Speicher des C128 von $1300 (4864) bis $1B86 (7046) und $BF00 (48896) bis $FF00 (65280) in BANK 0. Diese BASIC-Erweiterung unterstützt die Fähigkeiten des VDC 8563.
Dabei behindert es, auch wenn es im Speicher steht, in keiner Weise die Funktionen des VIC-II-Chips für die 40-Zeichen-Grafik oder andere Fähigkeiten Ihres C128. Lediglich könnten bei Floppy-Operationen dann Schwierigkeiten entstehen, wenn Sie entsprechende Befehle abgekürzt eingegeben haben. Am besten schreiben Sie diese Anweisungen alle aus.

DIE NEUEN BEFEHLE VON VDC-BASIC

old
Stellt nach einem New oder Reset ein BASIC-Programm wieder her, wenn das Programm weder physikalisch gelöscht, noch der Zeiger auf den Anfang des BASIC-Programmes geändert wurde.

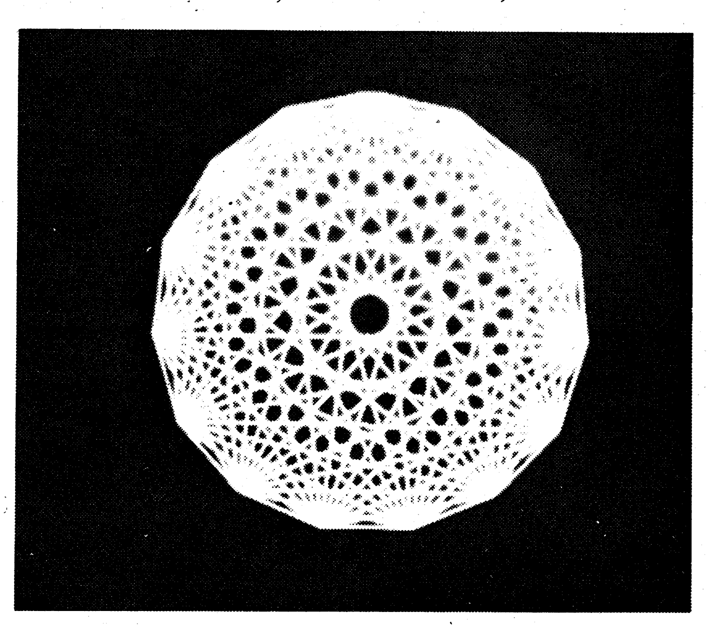
find
Zum Beispiel find Begriff oder find "Begriff" listet alle Zeichen auf dem Bildschirm, in denen der Begriff hinter find vorkommt. Allerdings, Suchbegriffe, die in Ihrem Programm als String stehen, also in Anführungszeichen, müssen auch bei der Find-Anweisung in solchen stehen. Falls Sie einen Text suchen, der in Anführungszeichen steht, fährt der Computer in der Zeile nach dem zweiten Anführungszeichen fort.

off
Damit wird VDC-BASIC abgeschaltet und der belegte RAM-Bereich wieder freigegeben.

screen
POKE für das VDC-RAM.

inscr
PEEK für das VDC-RAM.

wvreg
Beschreibt das angegebene VDC-Register mit dem von Ihnen angegebenen Wert, beispielsweise WVREG 25, 128 (schaltet die hochauflösende Grafik im 80-Zeichen-Bildschirm ein). Folgende Befehlssequenz wird dadurch ersetzt: POKE 54784,25: POKE 54785,128



rvreg
Gibt den Inhalt eines Registers aus.
Nach unserem vorstehenden Beispiel ergibt ein rvreg 25 die Zahl 128.

FUNKTIONEN

low (Zahl)
Liefert das Low-Byte eines Wertes.

high (Zahl)
Zeigt das High-Byte an.

DIE GRAFIK-BEFEHLE

gron
Schaltet den hochauflösenden Grafik-Modus (640 x 200 Bildpunkte) im 80-Zeichen-Bildmodus ein. Der Bildschirm wird gleichzeitig gelöscht.

groff
Mit diesem Befehl wird der 80-Zeichen-Grafikmodus wieder ausgeschaltet, der Zeichensatz erneut kopiert und der Bildschirm gelöscht. Sie befinden sich dann wieder im 80-Zeichen-Textmodus (so wie Sie es nach dem Einschalten Ihres C128 gewöhnt sind).

Folgende neue Befehle haben dieselbe Schreibweise (=Syntax) wie die bekannten des BASIC 7.0 im 40-Zeichen-Modus, allerdings haben Sie jetzt 640 Pixel in der horizontalen Richtung zur Verfügung. Lediglich beim
Bitte lesen Sie weiter auf Seite 37

```
10 rem vdc-basic.128                  <cg>
20 rem by martin ilse                 <eb>
30 rem (c) commodore-welt             <ld>
40 rem 80-zeichenmodus mit            <bd>
50 rem rgb-monitor                    <na>
60 rem ====================           <bl>

100 fori=4864to7046:readd$:pokei,d
ec(d$):next:sys4864:new               <lm>
101 dataa9,ff,85,fe,a9,f8,85,fd,a9
,fd,8d,b9,02,a0,01,a9,15,a2,7f,20,
af,02,88,a2,7f,a9,0f,20,af,02,8c,1
2,12,20,24,ca,a2,bf,8e,13,12,20,7d
,ff,0d,13,13,0b,0e,05                 <ho>
102 data93,20,d6,c4,c3,20,2d,20,c2
,41,53,49,43,20,28,30,38,2e,30,32,
2e,31,39,38,38,29,20,d6,20,31,2e,3
2,0d,00,a2,05,bd,6a,13,9d,0c,03,ca
,10,f7,a9,bb,a2,13,8d                 <ep>
103 datafc,02,8e,fd,02,60,70,13,96
,13,ac,13,85,fc,a9,13,a0,ca,20,e2,
43,90,0e,a2,00,29,7f,09,40,4c,b2,4
3,a5,fc,4c,26,43,a9,14,a0,26,20,e2
,43,90,f2,a2,ff,d0,e7                 <bj>
104 data29,3f,09,80,e8,f0,08,aa,a9
,13,a0,ca,4c,6a,51,aa,a0,26,a9,14,
d0,f6,29,3f,0a,a8,b9,3e,14,48,b9,3
d,14,48,4c,80,03,29,3f,0a,a8,b9,60
,14,48,b9,5f,14,48,4c                 <aj>
105 data56,79,56,4c,4f,43,41,54,c5
,4f,4c,c4,46,49,4e,c4,4f,46,c6,56,
42,4f,d8,53,43,52,45,45,ce,56,43,4
f,4c,4f,d2,47,52,4f,ce,47,52,43,4c
,d2,47,52,4f,46,c6,57                 <lc>
106 data56,52,45,c7,56,44,52,41,d7
,56,43,49,52,43,4c,c5,56,44,43,53,
41,56,c5,56,44,43,4c,4f,41,c4,00,d
4,ce,d4,2d,d3,4f,46,54,57,4f,52,4b
,00,49,4e,53,43,d2,56                 <pb>
107 data52,44,4f,d4,52,56,52,45,c7
,4c,4f,d7,48,49,47,c8,00,59,1a,69,
14,f6,1a,31,15,2b,19,71,14,0c,1a,4
e,15,5d,15,86,15,9e,15,b3,15,75,18
,94,1a,b2,1a,00,d4,ce                 <mn>
108 datad4,ca,14,64,1a,9f,14,8f,14
,97,14,00,a0,01,98,91,2d,4c,e5,5e,
20,03,88,86,fc,a5,17,c9,40,b0,10,8
d,32,17,a5,16,8d,33,17,20,d7,17,a5
,fc,4c,4a,1a,4c,28,7d                 <hf>
109 data20,b4,14,a5,16,4c,e3,14,20
,b4,14,a5,17,4c,e3,14,20,b4,14,a5,
17,d0,21,a5,16,c9,25,b0,1b,aa,20,5
4,1a,4c,e3,14,68,aa,68,a8,a5,17,48
,a5,16,48,98,48,8a,48                 <ek>
110 data20,da,77,4c,15,88,4c,28,7d
,20,b4,14,a5,17,c9,40,b0,f4,c6,16,
a6,16,e8,d0,02,c6,17,20,fb,14,20,f
b,14,a8,68,85,16,68,85,17,c0,00,f0
,03,4c,d4,84,a2,04,95                 <ln>
111 data63,ca,10,fb,68,68,60,a2,13
,a5,16,20,4c,1a,a2,12,a5,17,20,4c,
1a,20,52,1a,4c,52,1a,a2,ff,78,9a,d
8,20,42,e2,20,09,e1,20,93,e0,20,56
,e0,20,00,c0,20,7a,41                 <pi>
112 data20,51,42,20,45,40,20,00,13
,4c,2f,40,20,80,03,20,51,42,a2,ff,
8e,13,12,a9,e2,8d,10,13,a9,24,8d,1
a,13,a9,4c,8d,1b,13,4c,00,13,a2,19
,20,54,1a,09,80,29,bf                 <ho>
113 data8e,37,17,20,4c,1a,20,61,15
,a2,18,20,54,1a,29,7f,20,4c,1a,a2,
13,a9,00,20,4c,1a,a0,40,a2,12,98,2
0,4c,1a,a9,00,20,4a,1a,ca,20,4c,1a
,88,10,ee,60,a2,00,8e                 <ii>
114 data37,17,a2,19,20,54,1a,09,40
,29,7f,20,4c,1a,20,0c,ce,20,de,6a,
60,20,f4,87,e0,25,b0,0b,86,fc,20,0
9,88,8a,a6,fc,4c,4c,1a,4c,28,7d,a9
,01,8d,36,17,20,2e,18                 <ip>
115 dataf0,08,20,86,03,d0,03,4c,e6
,16,20,86,03,c9,2c,f0,05,c9,a4,f0,
01,60,48,20,80,03,a2,04,20,70,9e,8
6,fc,68,10,06,20,f4,15,4c,c6,15,20
,f2,9d,20,6a,18,a6,fc                 <ei>
116 data20,e6,16,4c,c6,15,20,6a,18
,a9,00,8d,24,17,8d,25,17,8d,26,17,
ad,2c,17,ac,2d,17,38,ed,28,17,aa,9
8,ed,29,17,a8,08,8a,28,20,9c,9d,8d
,1a,17,8c,1b,17,10,03                 <eb>
117 dataee,24,17,ad,2e,17,ac,2f,17
,38,ed,2a,17,aa,98,ed,2b,17,a8,08,
8a,28,20,9c,9d,8d,1c,17,8c,1d,17,1
0,03,ee,25,17,cc,1b,17,90,0c,d0,05
,cd,1a,17,90,05,a9,02                 <nh>
118 data8d,26,17,ad,26,17,49,02,8d
,27,17,aa,bd,1a,17,bc,1b,17,0a,8d,
1e,17,98,2a,8d,1f,17,a8,ad,1e,17,a
e,26,17,38,fd,1a,17,8d,22,17,98,fd
,1b,17,8d,23,17,a8,ad                 <gj>
119 data22,17,38,fd,1a,17,8d,20,17
,98,fd,1b,17,8d,21,17,4c,b9,16,20,
e7,17,ad,23,17,30,3e,20,06,18,ad,2
0,17,ac,21,17,18,6d,22,17,8d,22,17
,98,6d,23,17,8d,23,17                 <ec>
120 data20,e6,16,ae,26,17,bd,28,17
,dd,2c,17,d0,d3,bd,29,17,dd,2d,17,
d0,cb,20,f2,9d,ad,36,17,d0,01,60,a
9,00,8d,36,17,4c,e6,16,ad,1e,17,ac
,1f,17,4c,a8,16,20,40                 <fl>
121 data17,ad,6b,11,0d,6c,11,f0,28
,20,f1,17,20,40,17,4c,fa,17,ad,29,
17,c9,03,b0,17,c9,02,d0,07,ad,28,1
7,c9,80,b0,0c,ad,2b,17,d0,07,ad,2a
,17,c9,c8,90,01,38,60                 <mh>
122 data00,00,00,00,00,00,00,00,00
,00,00,00,00,00,00,00,00,00,00,00,
```

```
00,00,00,00,00,00,00,00,00,00,7f,b
f,df,ef,f7,fb,fd,fe,20,fa,16,b0,d4
,ad,28,17,8d,31,17,29                    <ok>
123 data07,8d,34,17,ad,29,17,8d,30
,17,ad,2a,17,8d,33,17,a9,00,8d,32,
17,ad,33,17,0e,33,17,2e,32,17,0e,3
3,17,2e,32,17,6d,33,17,8d,33,17,90
,03,ee,32,17,a2,04,0e                    <di>
124 data33,17,2e,32,17,ca,d0,f7,a2
,03,18,4e,30,17,6e,31,17,ca,d0,f6,
ad,31,17,18,6d,33,17,8d,33,17,90,0
3,ee,32,17,20,d7,17,20,52,1a,ae,34
,17,a4,83,d0,18,ac,35                    <pi>
125 data17,f0,19,c0,02,f0,09,3d,1c
,9d,a8,f0,02,a0,01,60,5d,1c,9d,4c,
cf,17,1d,1c,9d,4c,cf,17,3d,38,17,4
8,20,d7,17,68,4c,4a,1a,a2,12,ad,32
,17,20,4c,1a,a2,13,ad                    <pj>
126 data33,17,4c,4c,1a,ad,26,17,d0
,1f,ae,24,17,d0,09,ee,28,17,d0,03,
ee,29,17,60,ad,28,17,d0,03,ce,29,1
7,ce,28,17,60,ac,26,17,d0,e1,ac,25
,17,d0,09,ee,2a,17,d0                    <ao>
127 data03,ee,2b,17,60,ad,2a,17,d0
,03,ce,2b,17,ce,2a,17,60,20,5c,79,
20,d7,77,4c,15,88,ae,37,17,d0,03,4
c,79,a0,a2,00,8e,35,17,e8,86,83,48
,20,86,03,f0,11,c9,2c                    <lp>
128 dataf0,0d,68,f0,07,20,86,03,c9
,a4,f0,17,20,f4,87,e0,03,90,03,4c,
28,7d,e0,02,d0,05,8e,35,17,a2,00,8
6,83,a9,01,60,a2,07,bd,31,11,9d,28
,17,ca,10,f7,60,a9,00                    <kh>
129 data20,2e,18,a2,1f,20,52,9e,20
,06,9e,8c,54,11,8d,55,11,20,06,9e,
8c,56,11,8d,57,11,b0,09,ad,54,11,8
d,56,11,ad,55,11,8d,57,11,20,06,9e
,8c,5c,11,8d,5d,11,20                    <mo>
130 data06,9e,8c,5e,11,8d,5f,11,20
,06,9e,85,77,98,a4,77,20,77,9a,a2,
2d,a0,2b,20,7c,9d,90,0e,a9,68,a0,0
1,20,70,9d,9d,31,11,98,9d,32,11,a2
,03,bd,54,11,9d,58,11                    <ok>
131 dataca,10,f7,a9,90,20,f3,9a,a2
,07,bd,54,11,9d,60,11,ca,10,f7,20,
50,67,20,f2,9d,a2,02,20,1e,9e,8a,d
0,03,4c,28,7d,8e,20,12,18,ad,20,12
,6d,5c,11,8d,5c,11,90                    <gm>
132 data03,ee,5d,11,a2,2d,a0,2b,20
,7c,9d,b0,09,20,50,67,20,f4,15,4c,
04,19,a0,2d,20,52,67,4c,f4,15,a9,0
0,20,2e,18,a2,1f,20,6d,9e,a2,2b,20
,52,9e,20,06,9e,8c,54                    <ea>
133 data11,8d,55,11,20,1c,9e,e0,02
,90,03,4c,28,7d,8e,6c,11,8a,48,20,
89,63,68,d0,1c,f0,03,20,0b,64,20,f
4,15,ad,4e,11,d0,f5,a2,04,bd,5b,11
,9d,30,11,ca,d0,f7,8e                    <lp>
134 data6c,11,60,ae,35,17,f0,08,a2
,00,8e,6c,11,4c,28,7d,a2,00,ad,49,
11,4a,90,02,a2,02,bd,60,11,8d,5a,1
1,bd,61,11,8d,5b,11,a9,00,a2,03,9d
,56,11,ca,10,fa,a2,07                    <jj>
135 databd,31,11,48,ca,10,f9,20,f4
,15,a2,00,68,9d,31,11,e8,e0,08,d0,
f7,ad,5a,11,d0,05,ce,5b,11,30,a3,c
e,5a,11,a2,25,a0,1b,ad,49,11,4a,90
,02,a0,19,a9,00,4a,48                    <eh>
136 data20,6d,9d,9d,31,11,98,9d,32
,11,68,90,02,09,a0,e8,e8,a0,19,4e,
49,11,90,02,a0,1b,2e,49,11,e0,27,f
0,dd,a2,06,0a,f0,bd,90,08,fe,31,11
,d0,03,fe,32,11,0a,ca                    <en>
137 dataca,10,f1,30,95,a2,1a,20,54
,1a,29,0f,85,fc,20,3a,1a,86,fd,20,
5c,79,20,3a,1a,18,8a,69,68,a2,23,2
0,4c,1a,a6,fd,bd,4c,6a,0a,0a,0a,0a
,05,fc,a2,1a,4c,4c,1a                    <pf>
138 data20,f4,87,e0,11,b0,06,e0,00
,f0,02,ca,60,4c,28,7d,a2,1f,20,45,
a8,4c,cc,cd,a2,1f,20,45,a8,4c,da,c
d,ae,37,17,d0,03,4c,79,a0,4c,58,69
,20,f7,87,e0,02,90,10                    <kj>
139 dataf0,03,4c,28,7d,20,e4,1a,90
,02,a0,00,a9,00,f0,0a,8a,0a,aa,bd,
28,17,a8,bd,29,17,c0,00,d0,07,c9,0
0,d0,03,4c,f1,14,18,4c,c9,84,20,d1
,1a,20,d8,cd,a2,3f,8e                    <nm>
140 data00,ff,91,fc,a2,00,8e,00,ff
,c8,d0,ee,e6,fd,a5,fd,c9,ff,d0,e6,
60,20,d1,1a,a2,3f,8e,00,ff,b1,fc,a
2,00,8e,00,ff,20,ca,cd,c8,d0,ee,e6
,fd,a5,fd,c9,ff,d0,e6                    <np>
141 data60,a9,bf,85,fd,a9,00,85,fc
,a2,12,20,4c,1a,e8,20,cc,cd,a8,60,
a2,01,8e,35,17,ca,86,83,20,fa,16,9
0,01,60,20,e6,16,18,60,20,83,1b,f0
,4f,a2,00,86,fb,c9,22                    <bi>
142 datad0,11,e6,fb,20,7d,1b,c9,22
,d0,08,a4,fb,f0,04,20,80,03,98,9d,
00,0b,e8,c9,00,d0,e9,a5,2e,a6,2d,a
0,01,d0,1d,20,98,55,a0,02,20,ec,42
,aa,c8,20,ec,42,20,25                    <ng>
143 data51,20,b5,4b,a0,00,20,ec,42
,aa,c8,20,ec,42,85,62,86,61,20,ec,
42,d0,12,60,c8,e8,bd,00,0b,f0,d2,2
0,ec,42,dd,00,0b,f0,f1,d0,1c,88,84
,fa,a0,04,a2,00,20,ec                    <lm>
144 data42,f0,cf,c9,22,d0,06,a9,01
,45,fa,85,fa,a5,fb,c5,fa,f0,da,c8,
d0,e6,60,e6,3d,d0,02,e6,3e,4c,c9,0
3,ff                                     <aj>
```

```
1 rem vdc-basic.demo                    <nf>
2 rem by martin ilse                    <oj>
3 rem (c) commodore welt                <gc>
10 fast:gron                            <hg>
11 pi=3.14159265                        <fg>
20 color6,7:vcolor6,1                   <ol>
30 vdraw1,0,100to639,100                <dl>
40 fori=.to89.5step.5:a=sin(i/180*
pi)*100:b=i*16/9:vdraw1,319-b,100-
a:vdraw1,b,100-a:vdraw1,639-b,100+
a:vdraw1,320+b,100+a                    <lj>
50 vdraw1,159.5-b,100-a:vdraw1,160
+b,100+a:vdraw1,479-b,100+a:vdraw1
,480+b,100-a:nexti                      <ci>
60 vdcsave:groff:printchr$(11)chr$
(14)chr$(5)chr$(147)"Zur Fortsetzu
ng Taste druecken.":getkeya$:gron:
vdcload                                 <ih>
70 poke208,.:do:vdraw2,639,100to0,
100:loopuntilpeek(208)>.                <mc>
80 getkeya$:grclr                       <mb>
90 vcolor2,1                            <kp>
100 poke208,.:a=90:do                   <gp>
110 fori=10to100step5:vcircle2,160
,100,i,i,,,i,a:vcircle2,480,100,i,
i,,,i,a:nexti                           <ip>
120 loopuntilpeek(208)>.                <nn>
130 getkeya$:grclr:vcolor 11,1          <jh>
140 poke208,.:do                        <od>
150 fori=.to360step10:vbox2,120,50
,520,150,i:vcolor1+i/24,1:nexti         <mo>
160 loopuntilpeek(208)>.                <ea>
170 getkeya$:grclr:vcolor12,1           <kb>
180 poke208,.:do                        <il>
190 i=.:do:i=i+i/10+1:vlocatei*3.2
,i:vbox2,+i,+i:forj=ito40:nextj:lo
opuntili>=200                           <li>
200 loopuntilpeek(208)>.                <ca>
210 getkeya$:grclr                      <bc>
220 vcolor2,1:color6,1                  <cm>
230 fori=.to1000                        <cb>
240 vdraw1,rnd(ti)*640,rnd(ti)*200      <bi>
250 nexti:vdraw0,14,16to10,15:a=vr
dot(0):b=vrdot(1):c=vrdot(2)            <fl>
260 fori=47to.step-.1:wvreg35,46+i
:wvreg34,47-i:nexti:fori=.to299:ne
xti                                     <cd>
270 fori=.to47step.1:wvreg35,46+i:
wvreg34,47-i:nexti:wvreg34,125:wvr
eg35,100                                <nm>
280 poke208,.:getkeya$:groff            <am>
290 printa,b,c                          <af>
```

Fortsetzung von Seite 34

ersten Parameter nach der Anweisung, mit der Sie normalerweise die Farbquelle einschalten (dies kann eine Zahl von 0 bis 4 sein) ergeben sich diese Änderungen:

Es sind nur Zahlen von 0 bis 2 möglich, wobei diese im einzelnen bedeuten:

0 = löschen, also unsichtbar zeichnen, mit der Hintergrundfarbe,
1 = sichtbar zeichnen, mit der Farbe des Vordergrundes,
2 = invertieren.

vdraw
Zum Beispiel VDRAW 1,0,100 TO 639,100
zeichnet eine Linie in der Bildschirmmitte vom linken bis zum rechten Bildschirmrand.

vcircle
Zum Beispiel VCIRCLE 1, 320, 100, 60
zeichnet einen Kreis mit Mittelpunkt in der Bildschirmmitte mit einem Radius von 60 Bildpunkten.

vbox
Zum Beispiel VBOX 1,50,50,100,100
zeichnet ein Rechteck.

vlocate
Zum Beispiel LOCATE 320,100
setzt den „grafischen Cursor“ (den Sie nie zu Gesicht bekommen) genau in die Bildschirmmitte.

vdcsave
Der Inhalt des VDC-RAMs in Bank 0 wird in den RAM-Bereich $BF00 bis $FF00 übertragen.
Nunmehr können Sie Ihre Grafik mit folgendem Befehl als binäres File auf Diskette speichern:

BSAVE“(Bildname)“,ON B0, P48896 TO P 65280

Ganz einfach wie bei der 40-Zeichen-Grafik kann so ein Bild-File wieder geladen werden:
BLOAD“(Bildname)“
Um es aber dann auch zu sehen, müssen Sie den nächsten Befehl eingeben:

vdcload
Der Inhalt des RAM-Bereichs $BF00 bis $FF00 wird in das VDC-RAM übertragen.

vcolor
Im Gegensatz zur Grafik im 40-Zeichen-Modus können Sie hier zwei Parameter angeben: Zeichenfarbe und Fading, den Helligkeitsgrad dieser Zeichenfarbe. Probieren Sie ein wenig mit dieser Einstellmöglichkeit herum, es lassen sich interessante Effekte damit erzielen, zum Beispiel VCOLOR 7, 5.

vrdot (0–2)
Arbeitet genauso wie bei der 40-Zeichen-Grafik. Damit können Sie die aktuellen Bildschirmposition des grafischen Cursors (0 = X-Richtung, 1 = Y-Richtung) und den für diese Position gültigen Code der Farbquelle (2 = Farbcode) ermitteln.

Wir könnten uns vorstellen, daß es bestimmt den einen oder anderen an Grafik interessierten Leser gibt, der sich mit Hilfe dieser BASIC-Erweiterung ein Mal- und Zeichenprogramm für den VDC-Modus programmiert, so wie es schon einige für den 40-Zeichen-Modus gibt.

Martin Ilse/hb □

BASIC - ERWEITERUNG FÜR DEN 128PC: BASIC 7.5

Zugegeben: Das BASIC 7.0, mit dem der C128 ausgerüstet ist, ist eins der besten Exemplare dieser Computersprache für die kleinen 8-Bit-Rechner von Commodore. Trotzdem sind noch einige Lücken im Befehlswortschatz geblieben, die dieses Erweiterungs-Tool schließt.

Das Programm Basic V. 7.5 starten. Nach dem Laden belegt es den Bereich 5181 ($143d) bis 7167 ($1bff).

Das Programm schreibt sich im Speicher dann an die benötigten Stellen und braucht dann noch den Speicher von 5184 ($1440) bis 7167 ($1bff) in Bank 0, von 996 ($3e4) bis 1007 ($3ef) in der erweiterten Zeropage sowie fünf Byte in der Zeropage 250 ($fa) – 254 ($fe), und als letztes den Bereich von 1024 ($400) bis 1535 ($5ff) in der Bank 1 vor den Variablen.

Somit kann dieses Programm immer betrieben werden, außer wenn die Zeiger auf diese BASIC-Erweiterung, auf eine andere Routine gesetzt werden oder die oben genannten Bereiche beschreibt (Zeiger: $2fc – $2fd und $30c – $311). Dies ist deshalb zutreffend, weil die oben angegebenen Bereiche nicht vom Betriebssystem genutzt werden (auch die 512 Byte in Bank 1 nicht durch Hochsetzen von einigen Zeigern).

Anzumerken wäre noch, daß diese Erweiterung aufgrund des Programmteils in Bank 1 bei der Selbstinitialisierung alle Variablen löscht. Und, daß der Bereich in der Zeropage ($fa–$fe) nur als Zwischen- und Hilfsspeicher für die verschiedenen Routinen genutzt wird und ansonsten frei bleibt.

Wichtig: Diese BASIC-Erweiterung ist Reset-fest; sie wird durch einen Reset nicht abgeschaltet. Das geht nur mit dem Befehl „Off" oder einer Unterbrechung der Stromversorgung des Computers.

DIE NEUEN BEFEHLE:

(Achtung, alle Variablenoperationen funktionen nur bei Variablen des gleichen Typs!)

exchange a$ and b$
Tauscht den Inhalt von zwei Variablen aus, ohne eine dritte Variable zu benutzen (fünf bis 42 mal schneller als eine entsprechende Operation mit let).

exfi. a$(1),a$(10),a$(50) (exchange field)
Tauscht den Inhalt der Variablen von a$(1) bis a$(10) mit den Variablen ab a$(50) aus, ohne for-next-Schleife und ohne Austauschvariable, aber nur bei ein eindimensionalen Feldern (fünf bis 1200 mal schneller als eine entsprechende Operation mit let.)

exfi. a$(1), a$(10), a$(50) (exchange field)
Tauscht den Inhalt der Variablen von a$(1) bis a$(10) mit den Variablen ab a$(50) aus, ohne for-next-Schleife und ohne Austauschvariable, aber nur bei eindimensionalen Feldern. (fünf bis 1200 mal schneller als eine entsprechende Operation mit let).

transfer x to y
Vergleichbar mit y = x : x = 0, nur schneller (zwei bis 12 mal schneller als eine entsprechende Operation mit let).

transfi. x(1),x(10) tox(30) (transfer field)
Wie transfer, nur werden die Variablen von x(1) bis x(10) ab x(30) abgelegt. Dadurch spart man eine For-Next-Schleife und viel Zeit. Allerdings muß noch gesagt werden, daß dieser Befehl nur bei eindimensionalen Feldern und bei Bereichen, die sich nicht überlappen, funktioniert (dritte Variable darf nicht bei den Feldnummern zwischen den ersten beiden liegen), da es sonst zu Übertragungsfehlern kommen kann, die ganze Arrays löschen. (Zwei bis 350 mal schneller als eine entsprechende Operation mit let.)

clear x
Löscht die Variable; 1,5 mal schneller als let.

clfi. a$(1)toa$(100) (clear field)
löscht die Variablen von a$(1) bis a$(100) ohne For-Next-Schleife und ist wesentlich schneller als let (1,5 bis 32 mal schneller als eine entsprechende Operation mit let).

merge „Programm",8
Syntax des Load-Befehls.

Bewirkt das Zuladen eines zweiten BASIC-Programmes zu dem im Speicher vorhandenen. Die Zeilennummern können nun mit Renumber verändert werden, denn sie müssen größer sein als die des 1. Programmes.

merge 2
Mit Merge 2 werden dann die beiden Programme verknüpft, so daß der erste Teil wieder zu sehen und zu nutzen ist.

find „Text" oder Befehl
Listet alle Zeilen, in denen der Text oder der Befehl hinter Find vorkommt.

Texte, die im Programm in Anführungszeichen vorkommen, müssen auch jetzt in Anführungsstrichen stehen, da sonst die Erweiterung nicht zwischen Befehl und Text unterscheiden kann und es so zu Falschaussagen kommt. Nach dem Find-Befehl darf kein weiterer Befehl in der BASIC-Zeile stehen, da dieser von der Find- Routine dann als Suchobjekt angesehen wird. (Außer bei Text in Anführungszeichen; hier wird nach dem zweiten Anführungszeichen in der BASIC-Zeile fortgefahren.)

Anmerkung: Der Find-Befehl benutzt zur Zwischenspeicherung der Suchelemente den Berreich von 2816 ($b00) bis 3071 ($bff), also den Kassettenpuffer und den Puffer für Boot-Operationen.

dump
Mit dem Befehl Dump werden alle benutzten Variablen angezeigt. Diese allerdings nur namentlich. Arrays werden mit Dimensionen angegeben.

old
Gegenbefehl zu New. Funktioniert nur, wenn seit dem New die Zeiger auf den Programmanfang nicht verändert oder das Programm physikalisch gelöscht wurden.

cuon (cursor on)
Schaltet den Cursor beim Programmodus ein. (Nützlich für Eingabeschleifen mit Get- oder Getkey-Befehlen.)

coff (cursor off)
Schaltet den Cursor aus.

locks (lock stop)
Schaltet die Stop-Taste und Stop + Restore aus. Nicht speichern, während Stop-Taste blockiert, da die Daten dabei verfälscht wurden.

unlocks (unlock stop)
Schaltet die Stop-Taste wieder ein.

llist
Listet das Listing im Speicher auf dem Drucker. Voreingestellt ist Geräteadresse 4, Sekundäradresse 7. Die Geräteadresse kann mit bank0:poke7049, Geräteadresse und die Sekundäradresse können mit bank0: poke7051, Sekundäradresse geändert werden.

ofinput
Ein normaler Input. Nur wird das Fragezeichen unterdrückt.

reset
Führt einen Reset durch.

1wind erste Zeile, letzte Zeile
Wie ein Window, nur daß man die Zeilen bestimmen kann, da diese Möglichkeit viel häufiger benötigt wird als mit gleichzeitiger Änderung des Zeilenrandes.

Der zweite Wert ist optional, das heißt bei Angabe nur eines Wertes umfaßt das Window auch nur eine Zeile. Die Zeilenränder (linker, rechter Rand) des aktuellen Windows werden nicht verändert.

off
Schaltet die BASIC-Erweiterung Basic V 7.5 ohne Reset ab. Dabei wird allerdings der Bereich in Bank 1 nicht freigegeben. Dieses geschieht erst bei dem nächsten Reset.

screen Adresse, Wert
Ein Bonbon für die Besitzer eines 80-Zeichen-Monitors: Ein modifiziertes Poke beschreibt das VDC-RAM, das normalerweise von BASIC nur sehr schwer zugänglich ist. Maximale Adresse 16384.

xkey
Der normale Key-Befehl wurde erweitert, so daß auch andere Funktionstasten wie RUN und HELP belegt werden können.

Bitte lesen Sie weiter auf Seite 41

BASIC 7.5 — 1 —

```
10 rem basic 7.5 ==========128 <bl>
20 rem (c) commodore welt   === <fd>
30 rem ======================== <ae>
40 rem (p) by martin ilse   === <dg>
50 rem                      === <ei>
60 rem                      === <oe>
70 rem version 7.0 40/80z.  === <eo>
80 rem c 128 + floppy       === <ii>
90 rem ======================== <aa>

100 fori=5196to7172:readd$:pok
ei,dec(d$):next:sys5196:new    <gd>
101 data4c,72,16,20,cd,04,8d,0
4,ff,b1,fa,aa,b1,49,91,fa,8a,9
1,49,88,10,f3,60               <jf>
102 data20,d3,04,c8,98,18,65,f
c,90,02,e6,fd,85,fc,a0,00,20,0
3,04,e6,fa,d0,02               <lm>
103 datae6,fb,e6,49,d0,02,e6,4
a,a5,fd,c5,fb,30,2b,d0,e7,a5,f
a,c5,fc,d0,e1,60               <pa>
104 data85,fa,84,fb,20,ed,04,c
0,02,d0,0d,a5,fa,85,4b,a5,fb,8
5,4c,20,d0,54,a0               <ag>
105 data02,8d,04,ff,a9,00,91,f
a,88,10,fb,60,20,cd,04,84,fe,2
0,03,04,a4,fe,4c               <oe>
106 data46,04,20,d3,04,c8,84,f
e,a4,fe,88,20,03,04,20,46,04,a
5,fd,c5,fb,30,dd               <be>
107 datad0,06,a5,fa,c5,fc,f0,d
5,a5,fa,18,65,fe,90,02,e6,fb,8
5,fa,a5,49,18,65               <fn>
108 datafe,90,02,e6,4a,85,49,4
c,78,04,20,cd,04,c8,84,fe,20,4
6,04,a5,4a,c5,fb               <li>
109 data30,ad,d0,06,a5,fa,c5,4
9,f0,a5,a5,fa,18,65,fe,90,02,e
6,fb,85,fa,4c,ae               <bk>
110 data04,20,de,04,4c,ed,04,2
0,de,04,a2,02,20,e0,04,4c,ed,0
4,a2,00,95,fa,94               <cj>
111 datafb,e6,3d,d0,02,e6,3e,4
c,af,7a,a0,02,a5,0f,d0,07,88,a
5,10,d0,02,a0,04               <op>
112 data60,00,d4,ce,d4,00,20,d
5,05,a5,30,85,62,a5,2f,4c,1b,0
5,20,a0,05,20,fb               <ff>
113 data05,a5,61,18,69,07,90,0
2,e6,62,85,61,c5,31,d0,eb,a6,6
2,e4,32,d0,e5,f0               <ng>
114 data3b,20,a0,05,a9,28,20,e
4,03,a0,04,20,00,43,85,fa,d0,0
3,20,fb,05,20,80               <jo>
115 data05,c6,fa,d0,f6,a9,29,2
0,e4,03,20,fb,05,a0,02,20,00,4
3,aa,c8,20,00,43               <hc>
116 data65,62,85,62,8a,18,65,6
```

```
1,90,02,e6,62,85,61,c5,33,d0,c
1,a5,62,c5,34,d0                <ci>
117 databb,a9,14,20,e4,03,a9,e
6,d0,60,20,32,8e,a9,41,8d,00,f
f,60,a9,77,20,d7                <fb>
118 data05,a5,fa,0a,69,04,a8,2
0,00,43,aa,88,20,00,43,ca,e0,f
f,d0,02,e9,01,20                <bh>
119 datae4,03,4c,d5,05,a0,01,2
0,00,43,85,fc,48,88,20,00,43,8
5,fb,0a,90,08,68                <gn>
120 data0a,90,30,a9,25,d0,09,6
8,0a,90,04,a9,24,d0,01,98,85,f
d,a5,fb,20,e4,03                <ig>
121 dataa5,fc,20,e4,03,a5,fd,f
0,13,4c,e4,03,a9,f0,8d,dd,05,a
2,0a,bd,dc,05,9d                <pg>
122 datae4,03,ca,10,f7,60,20,a
f,7a,4c,00,04,20,af,7a,60,29,7
f,20,df,90,a9,41                <pc>
123 data8d,00,ff,60,a9,2c,4c,e
4,03,a2,ff,78,9a,d8,20,42,e2,2
0,09,e1,20,93,e0                <mn>
124 data20,56,e0,20,00,c0,20,7
a,41,20,51,42,20,45,40,20,72,1
6,4c,2f,40,a9,ff                <bk>
125 data85,fe,a9,f8,85,fd,a9,f
d,8d,b9,02,a0,01,a9,16,a2,7f,2
0,af,02,88,a2,7f                <na>
126 dataa9,4f,20,af,02,8c,00,f
f,84,2f,84,31,84,33,84,fd,20,2
4,ca,a5,d7,d0,74                <bl>
127 data20,7d,ff,0d,93,0b,0e,0
5,c2,41,53,49,43,20,d6,20,37,2
0,2e,35,20,28,d3                <jm>
128 data4f,46,54,2d,c2,41,53,4
9,43,2d,c5,52,57,45,49,54,45,5
2,55,4e,47,29,0d                <pm>
129 data3c,43,3e,20,42,59,20,d
4,ce,d4,20,2d,20,53,4f,46,54,5
7,4f,52,4b,20,30                <dh>
130 data33,2e,30,39,2e,31,39,3
8,37,0d,31,32,31,38,35,33,20,c
2,41,53,49,43,20                <li>
131 datac2,59,54,45,53,20,46,5
2,45,45,00,4c,f9,17,20,af,7a,4
c,00,04,20,af,7a                <id>
132 data60,20,7d,ff,0d,93,0b,0
e,05,20,20,20,20,20,20,20,20,2
a,2a,2a,2a,2a,2a                <jg>
133 data2a,2a,20,20,20,20,20,2
0,c2,41,53,49,43,20,d6,20,37,2
e,35,20,28,d3,4f                <cj>
134 data46,54,2d,c2,41,53,49,4
3,2d,c5,52,57,45,49,54,45,52,5
5,4e,47,29,20,20                <lm>
135 data20,20,20,20,2a,2a,2a,2
a,2a,2a,2a,2a,0d,20,20,20,20,2
0,20,20,20,2a,2a                <fj>
136 data2a,2a,2a,2a,2a,2a,20,3
c,43,3e,20,42,59,20,d4,ce,d4,2
0,2d,20,53,4f,46                <bd>
137 data54,57,4f,52,4b,20,28,c
d,41,52,54,49,4e,20,c9,4c,53,4
5,29,20,30,33,2e                <dh>
138 data30,39,2e,31,39,38,37,2
0,2a,2a,2a,2a,2a,2a,2a,2a,0d,2
0,20,20,20,20,20                <fb>
139 data20,20,2a,2a,2a,2a,2a,2
a,2a,2a,20,20,20,20,20,20,20,2
0,31,32,31,38,35                <fe>
140 data33,20,c2,41,53,49,43,2
0,c2,59,54,45,53,20,46,52,45,4
5,20,46,4f,52,20                <bl>
141 data55,53,49,4e,47,20,20,2
0,20,20,20,20,2a,2a,2a,2a,2a,2
a,2a,2a,00,a9,06                <kn>
142 data85,30,85,32,85,34,a2,0
9,bd,0c,17,9d,e4,03,ca,10,f7,a
2,05,86,fe,b9,4f                <if>
143 data15,a2,41,20,af,02,c8,d
0,f5,c6,fe,ce,12,18,a5,fe,c9,0
3,d0,ea,a9,15,8d                <ic>
144 data12,18,20,f8,51,a2,05,b
d,44,18,9d,0c,03,ca,10,f7,a9,9
5,a2,18,8d,fc,02                <np>
145 data8e,fd,02,60,4a,18,70,1
8,86,18,85,fc,a9,18,a0,a4,20,e
2,43,90,0e,a2,00                <ph>
146 data29,7f,09,40,4c,b2,43,a
5,fc,4c,26,43,a9,19,a0,1d,20,e
2,43,90,f2,a2,ff                <bl>
147 datad0,e7,29,3f,09,80,e8,f
0,08,aa,a9,18,a0,a4,4c,6a,51,a
a,a0,1d,a9,19,d0                <oj>
148 dataf6,29,3f,0a,a8,b9,28,1
9,48,b9,27,19,48,4c,80,03,29,3
f,0a,a8,b9,6b,19                <af>
149 data48,b9,6a,19,48,4c,56,7
9,52,45,53,45,d4,4d,45,52,47,4
5,b2,4d,45,52,47                <pp>
150 datac5,4f,4c,c4,46,49,4e,c
4,43,55,4f,ce,43,4f,46,c6,45,5
8,43,48,41,4e,47                <oo>
151 datac5,45,58,46,49,ae,43,4
c,45,41,d2,43,4c,46,49,ae,54,5
2,41,4e,53,46,45                <fb>
152 datad2,54,52,41,4e,46,49,a
e,4c,4f,43,4b,d3,55,4e,4c,4f,4
3,4b,d3,44,55,4d                <ie>
153 datad0,4f,46,49,4e,50,55,d
4,4c,4c,49,53,d4,4f,46,c6,53,4
3,52,45,45,ce,4c                <bj>
154 data57,49,4e,c4,58,4b,45,d
9,43,47,4f,54,cf,00,49,4e,53,4
3,d2,55,53,45,c4                <hl>
```

```
155 data00,3c,ff,6e,19,7e,19,a
0,19,c7,1a,56,1b,5c,1b,b5,1b,b
1,1b,ad,1b,a9,1b                <al>
156 dataa5,1b,a1,1b,62,1b,68,1
b,bd,1b,a8,19,75,1b,d2,1b,f4,1
9,c7,19,15,1a,52                <ni>
157 data1a,00,3c,43,3e,20,42,5
9,20,d4,ce,d4,2d,53,4f,46,54,5
7,4f,52,4b,00,77                <em>
158 data1a,58,1a,00,a9,00,f0,0
b,85,2e,a9,00,85,2d,a9,00,8d,7
0,19,60,ad,70,19                <je>
159 datad0,0a,a5,2d,8d,76,19,a
5,2e,8d,70,19,ad,10,12,ac,11,1
2,38,e9,02,b0,01                <db>
160 data88,85,2d,84,2e,4c,2c,9
1,a0,01,98,91,2d,4c,e5,5e,48,a
5,15,09,40,85,15                <fa>
161 data68,20,62,56,48,a5,15,2
9,bf,85,15,86,fc,84,fd,20,98,5
5,68,a6,fc,a4,fd                <en>
162 data60,20,f4,87,e0,19,b0,4
4,86,fc,20,86,03,f0,0b,20,09,8
8,e0,19,b0,36,e0                <mc>
163 data00,d0,05,a6,fc,4c,ec,1
9,8a,38,e5,fc,90,27,a5,fc,85,e
5,86,e4,4c,50,c1                <cj>
164 data20,03,88,86,fc,20,45,a
8,a5,17,c9,40,b0,10,a2,12,20,c
c,cd,e8,a5,16,20                <ld>
165 datacc,cd,a5,fc,4c,ca,cd,4
c,28,7d,f0,08,20,f4,87,e0,0a,4
c,e7,60,a2,ff,a0                <gj>
166 data00,e8,a9,08,85,77,bd,0
0,10,f0,19,86,fc,85,78,a2,06,b
d,4c,1a,ca,d0,02                <jp>
167 data05,fc,20,69,92,8a,10,f
2,20,3f,61,a6,fc,e0,09,d0,d9,6
0,2c,30,20,59,45                <km>
168 data4b,58,20,12,88,4c,e2,5
9,ae,11,12,ad,10,12,38,e5,2d,e
9,02,b0,01,ca,a8                <jb>
169 data8a,e5,2e,d0,08,c9,00,d
0,04,c0,00,f0,35,4c,3c,79,a5,1
7,48,a5,16,48,20                <kk>
170 datada,77,20,15,88,20,45,a
8,a5,17,c9,40,b0,86,c6,16,a6,1
6,e8,d0,02,c6,17                <lo>
171 data20,b4,1a,20,b4,1a,a8,6
8,85,16,68,85,17,c0,00,f0,03,4
c,d4,84,a2,04,95                <ie>
172 data63,ca,10,fb,68,68,60,a
2,13,a5,16,20,cc,cd,a2,12,a5,1
7,20,cc,cd,20,d8                <oe>
173 datacd,4c,d8,cd,20,54,1b,f
0,4f,a2,00,86,fb,c9,22,d0,11,e
6,fb,20,4e,1b,c9                <ai>
174 data22,d0,08,a4,fb,f0,04,2
0,80,03,98,9d,00,0b,e8,c9,00,d
0,e9,a5,2e,a6,2d                <of>
175 dataa0,01,d0,1d,20,98,55,a
0,02,20,ec,42,aa,c8,20,ec,42,2
0,25,51,20,b5,4b                <og>
176 dataa0,00,20,ec,42,aa,c8,2
0,ec,42,85,62,86,61,20,ec,42,d
0,12,60,c8,e8,bd                <ai>
177 data00,0b,f0,d2,20,ec,42,d
d,00,0b,f0,f1,d0,1c,88,84,fa,a
0,04,a2,00,20,ec                <dj>
178 data42,f0,cf,c9,22,d0,06,a
9,01,45,fa,85,fa,a5,fb,c5,fa,f
0,da,c8,d0,e6,60                <ho>
179 datae6,3d,d0,02,e6,3e,4c,c
9,03,20,45,a8,4c,6f,cd,20,45,a
8,4c,9f,cd,a9,73                <og>
180 dataa2,1b,d0,04,a9,6e,a2,f
6,8e,29,03,8d,28,03,a9,01,60,2
0,45,a8,20,bd,ff                <io>
181 dataa9,40,a2,04,a0,07,20,b
a,ff,20,c0,ff,a2,40,20,c9,ff,2
0,86,03,20,e2,50                <ph>
182 data20,98,55,a9,40,20,c3,f
f,a9,00,85,ba,4c,c9,ff,a9,72,d
0,12,a9,65,d0,0e                <pe>
183 dataa9,a8,d0,0a,a9,42,d0,0
6,a9,14,d0,02,a9,00,8d,e8,03,4
c,e4,03,a2,07,bd                <oa>
184 datacb,1b,9d,e4,03,ca,10,f
7,30,f0,a9,41,8d,00,ff,4c,00,0
5,a2,05,bd,fa,1b                <lc>
185 data9d,0c,03,ca,10,f7,a9,e
2,8d,82,16,a9,24,8d,8c,16,a9,4
c,8d,8d,16,a9,78                <jj>
186 data8d,fc,02,a9,4c,8d,fd,0
2,4c,72,16,21,43,cd,51,a9,48,0
0,25,1c,64,00                   <go>
```

BASIC 7.5

Fortsetzung von Seite 39

cgoto Zeilennummer
Die Zeilennummer kann jeden Wert zwischen 0 und 65535 annehmen. Der Befehl cgoto wird nicht durch Renumber angepaßt. Die Zeilennummer kann auch als Variable oder Formel angegeben sein. Beispiel: cgoto a*2+h–3

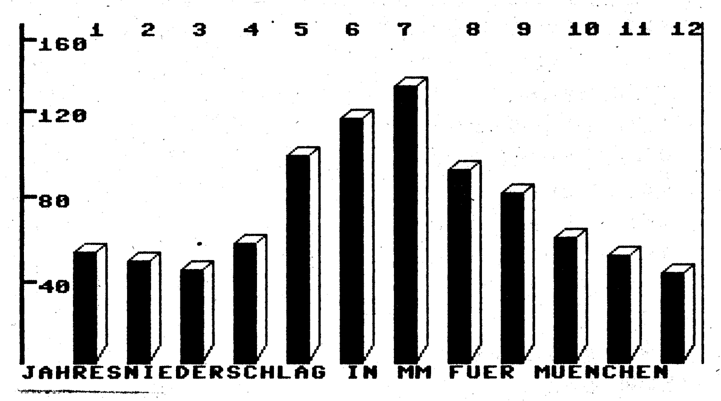
Funktionen:
inscr (Addresse)
Ein weiteres Bonbon für den 80-Zeichen-Benutzer: Ein modifiziertes Peek liest das VDC-RAM aus, das normalerweise von BASIC nur sehr schwer auszulesen ist. Maximale Adresse 16384.

used (dummy)
Gibt die Programmlänge des BASIC-Programms aus.

Martin Ilse ■

PC - GRAFIK

Neidvoll hat mancher Anwender die fantastischen grafischen Möglichkeiten eines großen Personal-Computers betrachtet, wenn's darum ging. Zahlenwerte in Windeseile als anschauliche Balken- oder Tortengrafik auf dem Bildschirm darzustellen. Mit dem folgenden Listing ist dies auch für den C 128 kein Problem mehr, wenn auch nur im 40-Zeichen-Modus.



Kurzbeschreibunng:
Nach dem Start des Programms landen Sie im Menü. Mit den Cursortasten werden die Menüpunkte angewählt und mit >RETURN< kommt man dann in den gewählten Bereich.
Nun zu den einzelnen Menüpunkten.

Anleitung:
Die Anleitung ist verhältnismäßig einfach gehalten, da sich das Programm in den einzelnen Menüpunkten von selbst erklärt und Bedienungsfehler so gut wie ausgeschlossen sind. Mit einem Tastendruck blättert man in der Anleitung weiter. Durch Drücken der Taste ‚m' gelangt man wieder ins Menü. Dabei spielt es keine Rolle, ob Groß- oder Kleinschreibung verwendet wird.

Balkengrafik:
Als erstes wird man nach der Anzahl der Eingaben gefragt, die auf zwölf begrenzt ist. Danach ist der Höchstwert anzugeben. Bei Werten, die im Bereich der Hunderttausender oder im Millionenbereich liegen, ist es günstiger, den Wert aufzuteilen. Anstelle von 800000 gibt man einfach nur 800 an und schreibt dann im Text eben ‚Werte in Tausend'. Der Text ist auf 38 Zeichen beschränkt. Bei einem längeren Text wird zurückgesprungen und man kann den Text gekürzt noch einmal eingeben. Bei Eingaben, die über oder unter den angegebenen Werten liegen, wird immer zurückgesprungen und zu einer neuen Eingabe aufgefordert. Nach einer Sicherheitsabfrage wird die Grafik aufgebaut. Mit dem Druck auf eine x-beliebige Taste kommt man ins Menü zurück.

Tortengrafik:
Nach der Abfrage zur Eingabenanzahl (auf zehn beschränkt), muß man sich entscheiden, ob man die Ausgabe in Absolut- oder Prozentwerten haben will. Nach der Eingabe der Werte kommt die übliche Sicherheitsabfrage. Wird sie mit j (für Ja) beantwortet, so wird die Grafik aufgebaut. Wie im Menüpunkt Balkengrafik geht's mit einem einfachen Tastendruck wieder ins Menü zurück. Bei n (für Nein) kann man seine Eingaben korrigieren.

Kurvengrafik:
Im Prinzip handhabt man die Kurvengrafik genauso wie die Balkengrafik. Der einzige Unterschied liegt in der Anzahl der Eingaben, die hier auf 100 begrenzt sind.

Grafik drucken:
Dieser Programmpunkt wurde offengelassen. Man kann den Punkt zwar anwählen, aber es erscheint nur der Hinweis auf Zeile 15000, ab der jeder seine eigene Hardcopy-Routine einbauen kann. Da fast jeder Drucker eine eigene Hardcopy-Routine vom Hires-Bildschirm benötigt, wurde auf eine Installation verzichtet.

Grafik ansehen:
Hier wird auf den Grafikbildschirm umgeschaltet. Mit einem Tastendruck geht's wieder zurück ins Menü.

Grafik laden:
Nach der Eingabe eines Namens für die Grafik wird sie von der Diskette geladen. Bei einem falschen Namen springt das Programm wieder in das Menü.

Grafik speichern:
Man wählt einen Namen für die Grafik (maximal sechzehn Zeichen), die sich im Speicher befindet, tippt in ein und dann >RETURN< und schon wird die Grafik gespeichert. Achtung! Auf der Diskette sollten noch mindestens 37 Blöcke frei sein, soviel Platz braucht nämlich eine Grafik.

Directory:
Mit Directory wird der Inhalt der Diskette aufgelistet. Zum Anhalten betätigt man die >NO SCROLL< -Taste und zum Weiterlaufen noch einmal. Es ist auch möglich, mit der Commodore-Taste das Scrollen zu verlangsamen. Sollte versehentlich die Stop-Taste gedrückt worden sein, so kann man mit dem Befehl CONT fortfahren.

Werte ansehen:
Um die zuletzt eingegebenen Werte aufzulisten, bedarf es dieses Menüpunktes. Die Handhabung ist die gleiche wie bei Directory.

Werte ändern:
Wenn dieser Punkt versehentlich aufgerufen wurde, so geht es mit der Eingabe einer Null wieder ins Menü. Ansonsten gibt man die Nummer des Wertes ein, den man ändern will. Dann wird der momentan gültige Wert angezeigt und man kann den neuen Wert eingeben. Sollte noch ein Wert geändert werden, so drückt man 'A'. Bei 'G' wird die Grafik mit dem neuen Wert aufgebaut. Mit Tastendruck geht's wieder in's Menü.

Ende:
Wenn man die Sicherheitsabfrage mit Nein beantwortet, kommt man wieder in das Menü, ansonsten wird das Programm beendet. *Josef Lindhuber*

```
10 rem pc-grafik===========128 <fh>
20 rem (p) commodore welt    == <hf>
30 rem ======================= <ae>
40 rem (c) by josef lindhuber= <el>
50 rem                      == <if>
60 rem                      == <nd>
70 rem version 7.0 40z/ascii== <nn>
80 rem c 128 + floppy       == <kf>
90 rem ======================= <aa>
95 gosub 60000                 <jp>
100 rem menue                  <ak>
110 :                          <bg>
120 clr                        <im>
130 dim f(11)                  <gf>
140 p=0:f(p)=1                 <cj>
150 printchr$(142)             <pe>
160 b$="balkengrafik    ":o$="
tortengrafik    ":k$="kurvengr
afik    ":d$=br$+"grafik druck
en  "+ye$                      <oi>
170 g$=br$+"grafik ansehen  "+
ye$:w$=gr$+"werte ansehen   "+
ye$:a$=bl$+"anleitung       "+
ye$:l$=cy$+"grafik laden    "+
ye$                            <cm>
180 s$=cy$+"grafik speichern"+
ye$:c$=cy$+"directory       "+
ye$:f$=gr$+"werte aendern   "+
ye$:e$=re$+"ende            "+
ye$                            <go>
190 :                          <lh>
200 color 0,1:color 4,1:color
5,6                            <ig>
210 scnclr                     <in>
220 char1,0,0,"
                         ",1   <hb>
230 char1,0,1,"            p c
 - g r a f i k           ",1   <ne>
240 char1,0,2,"
                         ",1   <ne>
250 color5,8                   <ek>
260 char,1,5,a$,f(0):char,20,5
,l$,f(6)                       <jb>
270 char,1,7,b$,f(1):char,20,7
,s$,f(7)                       <jh>
280 char,1,9,o$,f(2):char,20,9
,c$,f(8)                       <dc>
290 char,1,11,k$,f(3):char,20,
11,w$,f(9)                     <la>
300 char,1,13,d$,f(4):char,20,
13,f$,f(10)                    <ce>
310 char,1,15,g$,f(5):char,20,
15,e$,f(11)                    <hl>
320 char1,0,23,g3$+"mit cursor
tasten waehlen - dann >return<
",1                            <dg>
330 getkey m$                  <fk>
340 if m$=chr$(13) then onp+1g
oto510,1580,2380,3710,4580,467
0,4720,4940,5150,5220,5430,604
0                              <nk>
350 f(p) = 0                   <lp>
360 if m$=chr$(17) then begin  <ff>
370 ifp=5orp=11thenp=p-5:elsep
=p+1                           <ka>
380 bend                       <ak>
390 if m$=chr$(145)then begin  <ab>
400 ifp=0orp=6thenp=p+5:elsep=
p-1                            <id>
410 bend                       <id>
420 if m$=chr$(29) then begin  <ce>
430 ifp>5thenp=p-6:elsep=p+6   <cl>
440 bend                       <pl>
450 if m$=chr$(157)then begin  <lc>
460 ifp<6thenp=p+6:elsep=p-6   <bd>
470 bend                       <he>
480 f(p)=1                     <fp>
490 goto 260                   <gg>
500 end                        <gl>
510 :                          <dj>
520 rem --- anleitung ---      <bb>
530 :                          <gc>
540 scnclr                     <ce>
550 color0,1:color4,1:color5,1
5                              <ah>
560 printchr$(14)              <md>
570 forx=1to40:printrn$" "rf$;
:next                          <fi>
580 printrn$"           Anle
itung                "rf$      <gf>
590 forx=1to40:printrn$" "rf$;
:next                          <oc>
600 char1,0,23," m = Menue   j
ede andere Taste = weiter ",1  <io>
610 color5,16:window0,4,39,22,
1:print                        <lo>
620 printrn$"Allgemeines :"rf$
:print                         <nl>
630 print"Mit diesem Programm
koennen Sie Grafiken":print    <mp>
640 print"fuer die Statistik e
rstellen, speichern,":print    <gi>
650 print"laden, umaendern und
 ausdrucken.":print            <mb>
660 print"Wenn Sie bei Werte e
ine Null eingeben ":print      <ck>
670 print"kommen Sie wieder in
's Menue zurueck.":print       <pm>
680 print"Sollten Sie versehen
tlich >RUN/STOP< ge-":print    <hd>
690 print"drueckt haben, so ge
ben Sie 'cont' ein":print      <cp>
700 print"um im Programm fortz
ufahren."                      <ii>
710 getkeyw$                   <eg>
```

```
720 ifw$="m"orw$="M"then1570      <aj>
730 scnclr:print                  <jd>
740 printrn$"Balkengrafik :"rf
$:print                           <ni>
750 print"Sie koennen zwischen
 2 und 12 Balken ":print          <le>
760 print"waehlen. Den Maximal
wert koennen Sie ":print          <al>
770 print"frei bestimmen. Alle
rdings sollte er aus":print       <dj>
780 print"Gruenden der Uebersi
chtlichkeit nicht zu":print       <kk>
790 print"gross gewaehlt werde
n. Sie koennen ja im":print       <bg>
800 print"Text eine Angabe mac
hen, wie z. B. ":print            <fj>
810 print"'Werte in Tausend' o
der aehnlich."                    <jk>
820 getkeyw$                      <cp>
830 ifw$="m"orw$="M"then1570      <ig>
840 scnclr:print                  <af>
850 printrn$"Tortengrafik :"rf
$:print                           <oe>
860 print"Hier haben Sie die W
ahl zwischen der":print           <em>
870 print"Ausgabe in Absolut-
oder Prozentwerten. ":print       <nj>
880 print"Aus Gruenden der bes
seren Uebersicht-":print          <ko>
890 print"lichkeit sollten Sie
 die Werte nicht zu":print        <bo>
900 print"klein waehlen. Probi
eren Sie es einfach":print        <mn>
910 print"aus."                   <pf>
920 getkeyw$                      <hn>
930 ifw$="m"orw$="M"then1570      <ga>
940 scnclr:print                  <ck>
950 printrn$"Kurvengrafik :"rf
$:print                           <cd>
960 print"Bei mehr als 12 Wert
en benoetigen Sie":print          <bc>
970 print"die Kurvengrafik. Di
e Anzahl der Werte":print         <cp>
980 print"ist auf 100 begrenzt
.":print                          <le>
990 print"Zusaetzlich haben Si
e hier noch die ":print           <on>
1000 print"Moeglichkeit sich d
en Mittelwert Ihrer":print        <ch>
1010 print"Eingaben auf der Gr
afik ausgeben zu ":print          <ha>
1020 print"lassen."               <ag>
1030 getkeyw$                     <gc>
1040 ifw$="m"orw$="M"then1570     <mo>
1050 scnclr:print                 <jo>
1060 printrn$"Grafik drucken :
"rf$:print                        <nb>
1070 print"Hier koennen Sie in
 den Zeilen ab 15000":print       <oo>
1080 print"Ihre eigene Hardcop
y-routine einbauen.":print:pri
nt                                <kn>
1090 printrn$"Grafik ansehen :
"rf$:print                        <if>
1100 print"Bei diesem Menuepun
kt erscheint die":print           <lc>
1110 print"Grafik wieder auf d
em Bildschirm. Mit":print         <nf>
1120 print"dem Druck auf eine
beliebige Taste geht":print       <ji>
1130 print"es in's Menue zurue
ck."                              <aj>
1140 getkeyw$                     <ed>
1150 ifw$="m"orw$="M"then1570     <ii>
1160 scnclr:print                 <bb>
1170 printrn$"Grafik laden :"r
f$:print                          <mm>
1180 print"Um eine Grafik zu l
aden muessen Sie nur":print       <cg>
1190 print"den Namen der Grafi
k angeben. Vergessen":print       <ne>
1200 print"Sie nicht die Diske
tte einzulegen.":print:print      <pk>
1210 printrn$"Grafik speichern
 :"rf$:print                      <oj>
1220 print"Achten Sie darauf d
as auf der Diskette":print        <bg>
1230 print"noch mindestens 37
Blocks frei sind, ":print         <id>
1240 print"denn diesen Platz b
enoetigt eine Grafik."            <dk>
1250 getkeyw$                     <ci>
1260 ifw$="m"orw$="M"then1570     <gn>
1270 scnclr:print                 <ie>
1280 printrn$"Werte ansehen :"
rf$:print                         <jn>
1290 print"Mit der >NO"s2$"SCR
OLL< Taste koennen Sie ":print    <jm>
1300 print"bei laengeren Reihe
n die Werte anhalten":print       <mm>
1310 print"und durch erneutes
Druecken wieder ":print           <pk>
1320 print"loesen.":print         <mn>
1330 getkeyw$                     <cp>
1340 ifw$="m"orw$="M"then1570     <ig>
1350 scnclr:print                 <af>
1360 printrn$"Werte aendern :"
rf$:print                         <bn>
1370 print"Als erstes erschein
t der Text Ihrer ":print          <ng>
1380 print"Grafik. Sie koennen
 Ihn aendern oder ":print         <oo>
1390 print"einfach durch druec
ken von >Return<":print           <fl>
1400 print"uebernehmen. Dann w
erden Sie nach der ":print        <dk>
```

```
1410 print"Nummer des Wertes g
efragt, den Sie ":print         <be>
1420 print"aendern wollen. Sol
lten Sie die Nummer ":print     <cd>
1430 print"nicht wissen, sehen
 Sie einfach bei ":print        <fp>
1440 print"Werte ansehen nach.
"                               <ne>
1450 getkeyw$                   <li>
1460 ifw$="m"orw$="M"then1570   <ll>
1470 scnclr:print               <mj>
1480 printrn$"Directory :"rf$:
print                           <bl>
1490 print"Mit der >NO SCROLL<
 Taste koennen Sie":print       <pf>
1500 print"stoppen und wieder
loesen.":print                  <od>
1510 printrn$"Ende :"rf$:print  <ai>
1520 print"Nach einer Sicherhe
itsabfrage koennen ":print      <jg>
1530 print"Sie aussteigen oder
 in's Menue zurueck.":print:pr
int:print:print                 <na>
1540 printlr$"         Und nun
 viel Erfolg "g3$               <la>
1550 getkeyw$                   <ae>
1560 ifw$="m"orw$="M"then1570   <jo>
1570 window0,0,39,24,1:goto120  <op>
1580 :                          <jo>
1590 rem balkengrafik           <aj>
1600 :                          <mg>
1610 color0,1:color4,1:color1,
15:color5,4                     <la>
1620 clr:scnclr:graphic1,1:gra
phic0:cd=1:d2=0                 <hc>
1630 forx=1to40:printrn$" "rf$
;:next                          <cd>
1640 printrn$"             bal
kengrafik              "rf$     <co>
1650 forx=1to40:printrn$" "rf$
;:next                          <pp>
1660 print:print                <ih>
1670 printye$"anzahl der balke
n von 2 - 12 moeglich"cy$:prin
t:print:printchr$(7)            <ga>
1680 input"anzahl der balken";
e                               <lj>
1690 ife=0then120               <hh>
1700 ife<2ore>12thenprintc2$c2
$:printchr$(7):goto1680         <pm>
1710 print                      <ac>
1720 input"maximaler wert ";h   <al>
1730 ifh<.1ormw>10000000000000
00then1720                      <hc>
1740 print                      <do>
1750 ifh=50orh>50then1780       <hd>
1760 ifh<50then1770             <pm>
1770 v=h/4:z=int(v*100)/100:go
to1790                          <mm>
1780 z=int(h/4)                 <ld>
1790 z2=z*2:z3=z*3:z4=z*4       <eb>
1800 z$=str$(z):z2$=str$(z2):z
3$=str$(z3):z4$=str$(z4)        <oi>
1810 print"text (max. 38 zeich
en) :":print:print:print        <pj>
1820 print"  "zg$zg$zg$zg$zg$z
g$zg$zg$zg$zg$zg$zg$zg$zg$zg$z
g$zg$zg$zg$zg$zg$zg$zg$zg$zg$z
g$zg$zg$zg$zg$zg$zg$zg$zg$zg$z
g$zg$zg$:printc2$c2$c2$         <mm>
1830 inputt$                    <ek>
1840 color5,6:window0,12,39,24
,1                              <mn>
1850 l=len(t$)                  <bf>
1860 ifl>38thenprintchr$(7):go
to1780                          <cg>
1870 f=180/h                    <kh>
1880 s=e*25                     <lp>
1890 scnclr                     <lk>
1900 dima(e):dimi(e):dimw(e)    <id>
1910 fory=1toe                  <nd>
1920 i(y)=y*25                  <nf>
1930 print:printy". wert";      <he>
1940 inputw(y)                  <ac>
1950 ifw(y)>hthenprintchr$(7):
goto1930                        <pa>
1960 ifw(y)<0thenprintchr$(7):
goto1930                        <mb>
1970 nexty                      <ee>
1980 gosub6210                  <em>
1990 getkeyf$                   <he>
2000 iff$="j"then2100           <ng>
2010 iff$="n"then2030           <mb>
2020 goto1990                   <bc>
2030 printchr$(7):print:input"
welche nummer";n                <jg>
2040 ifn>ethen2030              <lj>
2050 ifn<1then2030              <cm>
2060 print:printn". eingabe ="
"w(n)                           <kh>
2070 print:input"neue eingabe"
;w(n)                           <bc>
2080 ifw(n)>hthen2070           <ol>
2090 goto1980                   <of>
2100 fory=1toe                  <kd>
2110 j$=str$(y)                 <ol>
2120 color1,y+1:graphic1        <fa>
2130 bb=w(y)*f                  <go>
2140 a(y)=190-bb                <hl>
2150 box1,i(y),a(y),i(y)+10,19
0,,1                            <dj>
2160 draw1,i(y),a(y)toi(y)+5,a
(y)-5toi(y)+15,a(y)-5toi(y)+10
,a(y)                           <bc>
2170 draw1,i(y)+15,a(y)-5toi(y
)+15,180toi(y)+10,190           <hd>
```

```
2180 char1,i(y)/8,0,j$                <ji>
2190 next                             <kp>
2200 width2:draw1,0,0to0,190:d
raw1,319,0to319,190:width1            <np>
2210 color1,16                        <je>
2220 char1,0,1,z4$                    <nb>
2230 char1,0,6,z3$                    <mh>
2240 char1,0,12,z2$                   <aj>
2250 char1,0,18,z$                    <bj>
2260 l$="O"                           <li>
2270 char1,0,1,l$                     <am>
2280 char1,0,6,l$                     <ej>
2290 char1,0,12,l$                    <gm>
2300 char1,0,18,l$                    <ol>
2310 draw1,1,0to1,190:draw1,31
9,190to319,0                          <ob>
2320 char1,0,24,t$                    <cp>
2330 :                                <ia>
2340 getkeyw$                         <je>
2350 graphic0:window0,0,39,24         <ig>
2360 ifd2=1then140                    <jh>
2370 goto130                          <fc>
2380 :                                <oe>
2390 rem tortengrafik                 <mk>
2400 :                                <am>
2410 scnclr:clr:cd=2                  <ml>
2420 color0,1:color4,1:color5,
6:color1,6                            <fa>
2430 printchr$(7)                     <pi>
2440 forx=1to40:printrn$" "rf$
;:next                                <ca>
2450 printrn$"                  to
rtengrafik              "rf$          <gh>
2460 forx=1to40:printrn$" "rf$
;:next                                <no>
2470 print:print:color5,6             <cl>
2480 print"anzahl der eingaben
 (2-10) :";                           <ja>
2490 inpute                           <fn>
2500 ife=0then120                     <ek>
2510 ife>10ore<2then2480              <hh>
2520 print:color5,4                   <mh>
2530 dima(e)                          <da>
2540 print:print:print                <ij>
2550 print"text (max. 38 zeich
en) :":print:print:print              <an>
2560 print"  "zg$zg$zg$zg$zg$z
g$zg$zg$zg$zg$zg$zg$zg$zg$zg$z
g$zg$zg$zg$zg$zg$zg$zg$zg$zg$z
g$zg$zg$zg$zg$zg$zg$zg$zg$zg$z
g$zg$zg$:printc2$c2$c2$               <ed>
2570 inputt$                          <id>
2580 window0,9,39,24,1                <hn>
2590 l=len(t$)                        <kj>
2600 ifl>38thenprintchr$(7):go
to2540                                <cm>
2610 window0,4,39,24,1                <jb>
2620 print:print:printe"eingab
en in absolutwerten  :":print:
color5,16                             <oh>
2630 print"ausgaben in "rn$"a"
rf$"bsolut oder "rn$"p"rf$"roz
ent"                                  <kj>
2640 getkeyag$                        <ne>
2650 ifag$="a"thenprintc2$+"au
sgabe in absolutwerten
        ":goto2680                    <bp>
2660 ifag$="p"thenprintc2$+"au
sgabe in prozenten
        ":goto3160                    <pi>
2670 printc2$c2$:goto2630             <pa>
2680 window0,10,39,24,1               <hb>
2690 color5,6                         <gd>
2700 fori=1toe                        <en>
2710 printi".";                       <da>
2720 inputw(i)                        <lm>
2730 s=s+w(i)                         <oo>
2740 nexti                            <ei>
2750 gosub6210                        <ok>
2760 getkeyf$                         <mg>
2770 iff$="j"then2880                 <ja>
2780 iff$="n"then2800                 <ck>
2790 goto2760                         <kj>
2800 printchr$(7):print:input"
welche nummer";n                      <le>
2810 ifn>ethen2800                    <pi>
2820 ifn<1then2800                    <dn>
2830 print:printn". eingabe =
"w(n)                                 <gm>
2840 s=s-w(n)                         <jf>
2850 print:input"neue eingabe"
;w(n)                                 <pm>
2860 s=s+w(n)                         <hb>
2870 goto2750                         <lj>
2880 color1,6:s$=str$(s)              <gj>
2890 v=s/360                          <fi>
2900 graphic1,1                       <db>
2910 box1,1,1,319,190                 <hk>
2920 char1,1,24,t$                    <oe>
2930 w1=0                             <oc>
2940 draw1,180,100to180,20            <jn>
2950 forj=1toe                        <bp>
2960 w2=w(j)/v                        <bf>
2970 w3=w2+w1                         <ob>
2980 s1$=str$(w(j))                   <fj>
2990 circle1,180,100,80,80,w1,
w3                                    <ph>
3000 cx=rdot(0):cy=rdot(1)            <fg>
3010 draw1,cx,cyto180,100             <oo>
3020 w4=w1+((w3-w1)/2)                <oo>
3030 circle0,180,100,70,70,w1,
w4                                    <io>
3040 cx=rdot(0):cy=rdot(1)            <mb>
3050 dx=rdot(0)/8:dy=rdot(1)/8        <af>
3060 draw0,cx,cyto180,100             <ph>
3070 char1,dx,dy,s1$,1                <hd>
```

```
3080 w1=w3                              <cb>
3090 nextj                              <mh>
3100 color1,13                          <eb>
3110 char1,1,1,"summe ="                <co>
3120 char1,8,1,s$                       <ki>
3130 getkeyw$:graphic0:window0
,0,39,24,1                              <gh>
3140 scnclr:ifd2=1then140               <ld>
3150 goto130                            <ef>
3160 color5,6:window0,10,39,24
,1:cd=3                                 <po>
3170 fori=1toe                          <bi>
3180 printi".";                         <ao>
3190 inputw(i)                          <ll>

3200 s=s+w(i)                           <lg>
3210 nexti                              <ki>
3220 gosub6210                          <pj>
3230 getkeyf$                           <ek>
3240 iff$="j"then3350                   <ag>
3250 iff$="n"then3270                   <jp>
3260 goto3230                           <jg>
3270 printchr$(7):print:input"
welche nummer";n                        <cl>
3280 ifn>ethen3270                      <jl>
3290 ifn<1then3270                      <mn>
3300 print:printn". eingabe =
"w(n)                                   <ln>
3310 s=s-w(n)                           <lo>
3320 print:input"neue eingabe"
;w(n)                                   <pn>
3330 s=s+w(n)                           <ej>
3340 goto3220                           <lc>
3350 color1,6:s$=str$(s)                <om>
3360 v=100/s                            <ge>
3370 graphic1,1                         <la>
3380 box1,1,1,319,190                   <dh>
3390 char1,1,24,t$                      <ch>
3400 w1=0                               <ga>
3410 draw1,200,100to200,20              <ba>
3420 forj=1toe                          <on>
3430 w2=w(j)*v*3.6                      <ai>
3440 w3=w2+w1:a$=str$(w(j))             <ok>
3450 s(j)=w(j)/s*100                    <ni>
3460 s3=s(j)*100:s1=(int(s3+0.
5))/100                                 <ak>
3470 s1$=str$(s1)                       <me>
3480 circle1,200,100,80,80,w1,
w3                                      <gf>
3490 cx=rdot(0):cy=rdot(1)              <ge>
3500 draw1,cx,cyto200,100               <jb>
3510 w4=w1+((w3-w1)/2)                  <gi>
3520 circle0,200,100,70,70,w1,
w4                                      <ja>
3530 cx=rdot(0):cy=rdot(1)              <ld>
3540 dx=rdot(0)/8:dy=rdot(1)/8          <ge>
3550 draw0,cx,cyto200,100               <ed>
3560 char1,dx,dy,s1$,1                  <gk>
3570 sp=len(s1$):char1,sp+dx,d
y,"%",1                                 <kl>
3580 w1=w3:s2=s2+s(j)                   <gf>
3590 color1,16                          <md>
3600 char1,0,j+2,s1$:char1,6,j
+2,"%=":char1,8,j+2,a$                  <cd>
3610 color1,6                           <am>
3620 nextj                              <bh>
3630 color1,13:s2$=str$(s)              <cb>
3640 char1,1,1,"summe ="                <nn>
3650 char1,8,1,s2$                      <bn>
3660 sn=len(s2$)                        <ec>
3670 char1,sn+9,1                       <ff>
3680 getkeyw$:graphic0:window0
,0,39,24,1                              <fa>
3690 ifd2=1then140                      <jf>
3700 goto130                            <mp>
3710 :                                  <fd>
3720 rem kurvengrafik                   <di>
3730 :                                  <hl>
3740 scnclr:clr:cd=4                    <ag>
3750 color0,1:color4,1:color5,
8:color1,8                              <fd>
3760 forx=1to40:printrn$" "rf$
;:next                                  <mb>
3770 printrn$"               kur
vengrafik              "rf$             <aj>
3780 forx=1to40:printrn$" "rf$
;:next                                  <pm>
3790 print:print:color5,6               <ai>
3800 print"anzahl der eingaben
 (2-100) :";                            <gf>
3810 inpute                             <bb>
3820 ife=0then120                       <fj>
3830 ife>100ore<2thenprintchr$
(7):goto3800                            <ic>
3840 print:color5,4                     <dc>
3850 dimw(e)                            <jk>
3860 print:print:print"maximal
er wert :";                             <ip>
3870 inputh                             <ah>
3880 ifh<.1orh>100000000000000
0then3860                               <bb>
3890 ifh=50orh>50then3920               <ja>
3900 ifh<50then3910                     <co>
3910 v=h/4:z=int(v*100)/100:go
to3930                                  <ae>
3920 z=int(h/4)                         <em>
3930 z2=z*2:z3=z*3:z4=z*4               <pd>
3940 z$=str$(z):z2$=str$(z2):z
3$=str$(z3):z4$=str$(z4)                <hi>
3950 print:print                        <fl>
3960 print"text (max. 38 zeich
en) :":print:print:print                <gm>
3970 print"  "zg$zg$zg$zg$zg$z
g$zg$zg$zg$zg$zg$zg$zg$zg$z
g$zg$zg$zg$zg$zg$zg$zg$zg$z
g$zg$zg$zg$zg$zg$zg$zg$zg$z
```

```
g$zg$zg$:printc2$c2$c2$            <ap>
3980 inputt$                       <lp>
3990 window0,10,39,24,1            <ei>
4000 l=len(t$)                     <fp>
4010 ifl>38thenprintchr$(7):go
to3950                             <nh>
4020 color5,6                      <nc>
4030 forx=1toe                     <op>
4040 printx". wert   ";            <od>
4050 inputw(x)                     <lh>
4060 ifw(x)>horw(x)<0thengoto4
040                                <fh>
4070 print                         <ie>
4080 next                          <ib>
4090 gosub6210                     <ac>
4100 getkeyf$                      <oc>
4110 iff$="j"then4210              <ka>
4120 iff$="n"then4140              <mc>
4130 goto4100                      <mn>
4140 printchr$(7):print:input"
welche nummer";n                   <ej>
4150 ifn>ethen4140                 <ik>
4160 ifn<1then4140                 <lg>
4170 print:printn". eingabe =
"w(n)                              <hk>
4180 print:input"neue eingabe"
;w(n)                              <pb>
4190 ifw(n)>hthen4180              <kf>
4200 goto4090                      <jc>
4210 print:print"mittelwert gr
afisch anzeigen (j/n)"             <lc>
4220 getkeyw$                      <ik>
4230 ifw$="j"then4260              <dm>
4240 ifw$="n"thenr=0:goto4300      <ph>
4250 goto4220                      <oi>
4260 u=0                           <io>
4270 fori=1toe:u=w(i)+u:next       <nj>
4280 j=u/e                         <dm>
4290 d=190/h:r=190-(j*d)           <pl>
4300 window0,0,39,24,1             <dk>
4310 as=317/(e-1)                  <df>
4320 d=190/h                       <mo>
4330 c=1                           <hf>
4340 graphic1,1                    <id>
4350 p=1:q=190-(w(1)*d)            <nd>
4360 forx=1toe                     <nm>
4370 l=w(x)*d                      <eg>
4380 draw1,c,190toc,190-l          <il>
4390 draw1,p,qtoc,190-l            <bm>
4400 p=rdot(0):q=rdot(1)           <pc>
4410 c=c+as                        <oi>
4420 next                          <cm>
4430 draw1,1,rto319,r              <jc>
4440 box1,0,0,319,190:box1,1,0
,318,190                           <hp>
4450 char1,0,0,z4$                 <lm>
4460 char1,0,6,z3$                 <pc>
4470 char1,0,12,z2$                <al>
4480 char1,0,18,z$                 <cj>
4490 l$="O"                        <ii>
4500 char1,0,0,l$                  <hm>
4510 char1,0,6,l$                  <mk>
4520 char1,0,12,l$                 <ib>
4530 char1,0,18,l$                 <gf>
4540 color1,6:char1,1,24,t$        <dk>
4550 getkeyw$:graphic0             <jo>
4560 ifd2=1then140                 <ne>
4570 scnclr:goto130                <dp>
4580 :                             <cf>
4590 rem grafik drucken            <bh>
4600 :                             <em>
4610 scnclr                        <ba>
4620 color5,6                      <hk>
4630 char1,7,11,"UCCCCCCCCCCC
CCCCCCCCCI"                        <mi>
4640 char1,7,12,"B grafik wird
 gedruckt B"                       <gi>
4650 char1,7,13,"JCCCCCCCCCCC
CCCCCCCCCK"                        <am>
4660 goto6300                      <fo>
4670 :                             <nk>
4680 rem grafik ansehen            <dn>
4690 :                             <ac>
4700 graphic1                      <kg>
4710 getkeyw$:graphic0:f(p)=0:
goto140                            <ic>
4720 :                             <dp>
4730 rem grafik laden              <lp>
4740 :                             <gh>
4750 scnclr                        <cj>
4760 color4,10:color5,8            <fj>
4770 char1,0,1,"
                         ",1       <nk>
4780 char1,0,2," grafik laden
                         ",1       <ia>
4790 char1,0,3,"
                         ",1       <ln>
4800 color5,6                      <mh>
4810 char1,2,22,"_ = menue"        <nh>
4820 char1,21,9,"^^^^^^^^^^^^^
^^^"                               <gi>
4830 char1,2,8,"name der grafi
k"                                 <bd>
4840 input"  ";g$                  <bb>
4850 ifg$="_"thenf(p)=0:goto14
0                                  <bc>
4860 char1,2,8,"
                         "         <ab>
4870 char1,2,8,g$                  <nl>
4880 char1,2,10,"wird geladen"
:sleep1                            <bm>
4890 trap6230                      <cd>
4900 graphic1,1                    <la>
4910 bload(g$),p7168               <mb>
4920 printchr$(7):getkeyw$:gra
phic0:f(p)=0                       <jj>
```

```
4930 goto140                              <pg>
4940 :                                    <pi>
4950 rem grafik speichern                 <kg>
4960 :                                    <ca>
4970 scnclr                               <oc>
4980 color4,11:color5,10                  <hf>
4990 char1,0,1,"
                                   ",1    <kf>
5000 char1,0,2," grafik speich
ern                                ",1    <nh>
5010 char1,0,3,"
                                   ",1    <ic>
5020 color5,6                             <jf>
5030 char1,2,22,"_ = menue"               <pj>
5040 char1,21,9,"^^^^^^^^^^^^^
^^^"                                      <mn>
5050 char1,2,8,"name der grafi
k"                                        <be>
5060 input"  ";g$                         <fj>
5070 ifg$="_"thenf(p)=0:goto14
0                                         <jf>
5080 l=len(g$)                            <bk>
5090 ifl>16thenprint:print:pri
ntwh$"   name zu lang"gr$:prin
tchr$(7):sleep2:goto4970                  <kp>
5100 char1,2,8,"
                                   "      <cf>
5110 char1,2,8,g$                         <dk>
5120 char1,2,10,"wird gespeich
ert"                                      <kd>
5130 bsave(g$),d0,on b0,p7168t
op16384                                   <pa>
5140 f(p)=0:goto140                       <og>
5150 :                                    <jo>
5160 rem directory                        <ab>
5170 :                                    <mg>
5180 scnclr:directory                     <ie>
5190 color5,6:print:print"tast
e druecken"                               <jh>
5200 getkeyw$                             <ab>
5210 f(p)=0:goto140                       <gk>
5220 :                                    <ck>
5230 rem werte ansehen                    <ne>
5240 :                                    <fd>
5250 color0,1:color4,12:color5
,16                                       <he>
5260 scnclr                               <cj>
5270 forx=1to40:printrn$" "rf$
;:next                                    <hm>
5280 printrn$"               werte
 ansehen               "rf$               <pc>
5290 forx=1to40:printrn$" "rf$
;:next                                    <mb>
5300 printrn$" anhalten mit >n
o scroll - taste<      "rf$               <ap>
5310 forx=1to40:printrn$" "rf$
;:next                                    <pm>
5320 window0,6,39,24                      <kk>
5330 color5,13:print:printt$              <ck>
5340 window0,8,39,24                      <id>
5350 print:color5,6                       <ap>
5360 foras=1toe                           <ka>
5370 printas". wert"w(as):slee
p1                                        <dg>
5380 next                                 <le>
5390 color5,16                            <mf>
5400 print:printrn$"  taste

"rf$                                      <ia>
5410 getkeyw$                             <cp>
5420 window0,0,39,24:f(p)=0:go
to140                                     <ch>
5430 :                                    <na>
5440 rem werte aendern                    <mc>
5450 :                                    <pi>
5460 color0,1:color4,12:color5
,10                                       <lg>
5470 scnclr                               <mo>
5480 forx=1to40:printrn$" "rf$
;:next                                    <em>
5490 printrn$"               werte
 aendern               "rf$               <mj>
5500 forx=1to40:printrn$" "rf$
;:next                                    <ca>
5510 char1,0,23," 0 = menue
                              ",1         <lf>
5520 window0,4,39,22                      <he>
5530 color5,6                             <jf>
5540 ifcd=1orcd=4then5560                 <ni>
5550 ifcd=2orcd=3then5830                 <ob>
5560 color5,13:scnclr:print:pr
intt$                                     <al>
5570 printc2$c2$:open1,0:input
#1,t$:close1:print:color5,9               <aa>
5580 window0,7,39,20                      <im>
5590 print:print" "e"werte":pr
int:print" maximalwert ="h:pri
nt:color5,6                               <ce>
5600 print:print" welchen wert
 wollen sie aendern ?":print:p
rint                                      <lo>
5610 inputae                              <nj>
5620 ifae>ethen5610                       <po>
5630 ifae<1then6030                       <mn>
5640 print:print" "ae". wert =
 "w(ae):print:print                       <ih>
5650 print"  neuer wert :":pri
nt                                        <gd>
5660 inputw(ae)                           <ji>
5670 ifw(ae)>hthenprintchr$(7)
:goto5660                                 <gp>
5680 ifw(ae)<0thenprintchr$(7)
:goto5660                                 <ap>
5690 print:printrn$" a = aende
rn               g = grafik
"rf$                                      <hg>
```

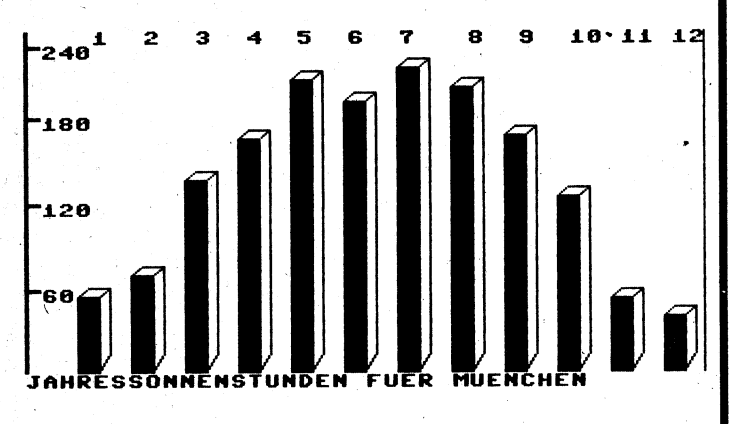
```
5700 color5,16                          <ki>
5710 print:printrn$"  taste

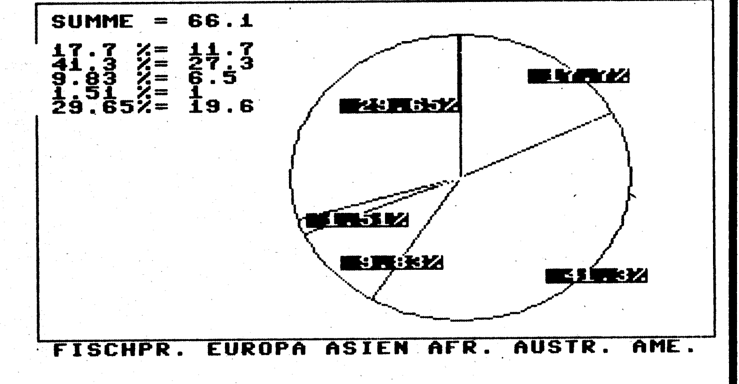
"rf$                                    <ef>
5720 getkeyw$                           <ka>
5730 ifw$="a"then5590                   <gg>
5740 ifw$="g"then5760                   <hp>
5750 goto5720                           <eb>
5760 window0,0,39,24:f(p)=0             <fd>
5770 d2=1:color4,1:graphic1,1           <kb>
5780 ifcd=1goto2100                     <ch>
5790 ifcd=2goto2880                     <mi>
5800 ifcd=3goto3350                     <nb>
5810 ifcd=4goto4260                     <kp>
5820 goto140                            <mo>
5830 color5,13:scnclr:print:pr
intt$                                   <gm>
5840 printc2$c2$:open1,0:input
#1,t$:close1:print:color5,9             <gg>
5850 window0,7,39,20                    <dp>
5860 print:print" "e"werte":pr
int:color5,6                            <pj>
5870 print:print" welchen wert
 wollen sie aendern ?":print:p
rint                                    <ka>
5880 inputae                            <eo>
5890 ifae>ethen5880                     <nb>
5900 ifae<1then6030                     <in>
5910 print:print" "ae". wert =
 "w(ae):print:print                     <oe>
5920 s=s-w(ae)                          <da>
5930 print" neuer wert :":prin
t                                       <on>
5940 inputw(ae)                         <nc>
5950 s=s+w(ae)                          <em>
5960 print:printrn$" a = aende
rn                g = grafik
"rf$                                    <po>
5970 color5,16                          <im>
5980 print:printrn$"  taste

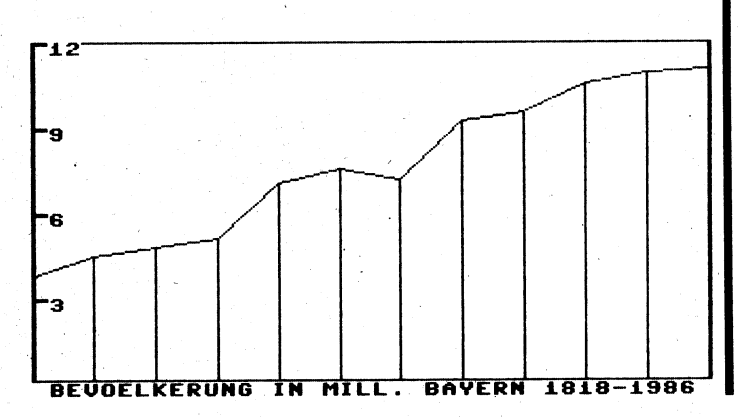
"rf$                                    <pe>
5990 getkeyw$                           <jd>
6000 ifw$="a"then5860                   <dd>
6010 ifw$="g"then5760                   <ii>
6020 goto5990                           <dn>
6030 window0,0,39,24:f(p)=0:go
to140                                   <og>
6040 :                                  <jj>
6050 rem ende                           <ep>
6060 :                                  <mb>
6070 color5,8:scnclr                    <ea>
6080 char1,2,11,"
                              ",1       <dc>
6090 char1,2,12," sie wollen a
ussteigen ???  (j/n) ",1                <cb>
6100 char1,2,13,"
                              ",1       <lc>
```

```
6110 getkeyw$                           <bk>
6120 ifw$="j"then6150                   <nk>
6130 ifw$="n"thenf(p)=0:goto14
0                                       <mn>
6140 goto6110                           <ad>
6150 scnclr                             <ce>
6160 forx=2to10step2:char1,x,x
,"auf wiedersehen":next                 <ln>
6170 color5,7                           <lo>
6180 forx=12to20step2:char1,x,
x,"(c)  eljot":next                     <nn>
6190 color5,6                           <pk>
6200 end                                <ca>
6210 rem unterprogramm                  <kg>
6220 printchr$(7):print:printy
e$"alles richtig ???"gr$:retur
n                                       <pi>
6230 :                                  <bg>
6240 rem diskettenfehler                <lb>
6250 :                                  <dp>
6260 graphic0:window0,0,39,24:
color0,3:color4,2:color5,8:pri
ntchr$(7):f(p)=0                        <cd>
6270 char1,0,10,"
diskettenfehler                ",1      <lc>
6280 sleep2                             <ef>
6290 goto140                            <fc>
6300 :                                  <kd>
6310 rem hardcopy-routine               <ng>
6320 :                                  <ml>
6330 color5,8                           <mj>
6340 print:print:print" hier k
onnen sie ihre eigene ":print           <kf>
6350 print" hardcopy-routine e
inbauen"                                <gn>
6360 sleep3                             <mi>
6370 printchr$(7):f(p)=0:goto1
40                                      <ij>
6380 :                                  <ec>
6390 rem =====================          <eh>
6400 rem da jeder drucker eine
 andere                                 <kp>
6410 rem hardcopy-routine beno
etigt                                   <hc>
6420 rem wurde hier auf den ei
nbau                                    <pn>
6430 rem verzichtet.                    <bd>
6440 rem so kann jeder seine e
igene                                   <pc>
6450 rem hardcopy-routine               <bl>
6460 rem implementieren                 <dg>
6470 rem =====================          <eg>
60000 rem nachspann ==========          <dg>
60010 rem farb/steuercodes              <gf>
60020 wh$=chr$(005):rn$=chr$(0
18)                                     <ic>
60030 re$=chr$(028):gr$=chr$(0
30)                                     <gd>
```

```
60040 bl$=chr$( 031) :c2$=chr$( 1
45)                           <ej>
60050 rf$=chr$( 146) :br$=chr$( 1
49)                           <mm>
60060 lr$=chr$( 150) :g3$=chr$( 1
55)                           <kp>
60070 ye$=chr$( 158) :cy$=chr$( 1
59)                           <of>
60080 rem zeichensatz/graphik <nj>
60090 s2$=chr$( 160) :zg$=chr$( 1
77)                           <dg>
60100 return                  <ol>
```







Kurzanleitung zu VDC-MON'

DEN 80-ZEICHEN-CHIP RICHTIG PROGRAMMIEREN

Der VDC (Video-Display-Controller) 8563 scheint eines der bestgehütesten Geheimnisse des Herstellers zu sein. Das folgende Programm lüftet es und gibt Ihnen Ihnen eine Menge Anregungen, um diesen Speicherbaustein zu manipulieren.

Nach dem Laden und Starten von ‚VDC-MON‘ erscheint ein Menü mit neun Punkten. Das Programm erklärt sich beim Ablauf weitestgehend selbst.

1. Register ändern
Nach Aufruf dieses Punktes fragt das Programm nach der zu ändernden Adresse. Wird diese eingegeben (0—36 Register), so erscheint der augenblickliche Inhalt und der Inhalt im Normalzustand. Den neuen Wert können Sie in einem der drei wichtigsten Zahlensysteme eingeben (Hexadezimal, Dezimal und Binär.

2. Register ändern (+/—)
Wählen Sie diesen Punkt an, so fragt das Programm, wie Punkt 1, nach der zu ändernden Adresse. Jetzt kann der Inhalt mit '+' oder '—' erhöht oder erniedrigt werden. Mit 'Return' verläßt man diesen Menüpunkt.

3. Register lesen
Geben Sie den Inhalt des auszulesenden Registers ein. Darauf zeigt Ihnen das Programm den Inhalt in allen drei (in Menüpunkt 1. aufgeführten) Zahlensystemen an.

4. Register zurücksetzen
Falls die Register so verstellt sind, daß der Bildschirm flackert oder nichts mehr zu sehen ist, so können Sie sie hier zurücksetzen. Mit der Space-Taste werden die links angezeigten Register in den Ausgangszustand versetzt.

5. VDC-Parameter laden
Mit diesem Punkt wird eine sequenteille Datei von Disk, die mit Punkt 8 erstellt wurde und Registerdaten enthält, sofort in den VDC-Speicher oder erst in den Zwischenspeicher geladen (je nach Punkt 8).

6. VCD-Parameter speichern
Die im Zwischenspeicher befindlichen Daten werden in einem Seq-File auf Disk gespeichert.

7. VDC mit eingegebenen Parametern füllen
Wenn Punkt 8 auf 'ERST NACH SCHRITT 7' steht, so werden mit Punkt 7 die Register aus dem Zwischenspeicher in den VDC-Speicher geschrieben.

8. Parameter in VDC speichern: SOFORT / ERST NACH SCHRITT 7
Bei Aufruf dieseys Punktes kann man einstellen, ob die geänderten Register sofort in den VDC oder erst in den Zwischenspeicher geschrieben werden (relevant bei Punkten 1,2,5).

9. Ende
Das Programm wird beendet.

Erklärung zu dem Zwischenspeicher
Nehmen wir an, daß Sie eine bestimmte Kombination neuer Registerinhalte haben und ausprobieren möchten. Sie müßten also die neuen Inhalte nacheinander in den VDC-Speicher schreiben. Doch oft kommt es vor, daß bei dem Beschreiben der Bildschirm zu flackern beginnt oder dunkel wird. Deshalb wäre es recht praktisch, wenn man die Inhalte erst zwischenspeichern könnte, um sie dann in den VDC zu kopieren. Diese Möglichkeit bietet Ihnen der Zwischenspeicher (siehe Punkte 5,7,8).

Benötigte Peripherie:

C128 mit 80-Zeichen-Bildschirm und Floppy 1541/70/71.

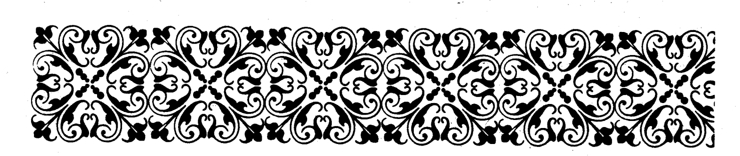
Die folgenden Tabellen sollen demonstrieren, was das Programm VDC-MON alles kann.
Eingaben nach Tabelle 1 ermöglichen den Interlace-Modus. Damit können alle Zeichen in halber Größe dargestellt werden.
Tabelle 2 versetzt den VDC-Chip in die Lage, Zeichen so zu vergrößern, als würden Sie am 40-Zeichen-Bildschirm dargestellt. *Harald Görl* ■

Tabelle 1

VDC-Register	Neuer Inhalt
8	255
4	76/77/78 (je nach Bildschirm)
1	78
27	2
7	49

Tabelle 2

VDC-Register	Neuer Inhalt
1	50
2	59
0	71
27	30
25	86



```
10 rem vdc-mon============128 <mj>
20 rem (p) commodore welt   == <hf>
30 rem ====================== <ae>
40 rem (c) by harald goerl  == <pn>
50 rem                      == <if>
60 rem                      == <nd>
70 rem version 7.0 80z/ascii== <on>
80 rem c 128 + floppy       == <kf>
90 rem ====================== <aa>

95 gosub 60000                 <jp>
100 fast                       <lc>
110 printcl$:dima(36),ax$(36),
in(36):fori=0to36:reada(i):sys
dec("cdcc"),a(i),i:in(i)=a(i):
readax$(i):next:zx=-1          <ia>
120 printcl$tab(35)wh$"vdc-mon
":printtab(35)"-------"        <ij>
130 printc4$c4$tab(20)"1) regi
ster aendern":printtab(20)c4$"
2) register aendern (+/-)":pri
nttab(20)c4$"3) register lesen
":printtab(20)c4$"4) register
zuruecksetzen"                 <ae>
140 printtab(20)c4$"5) vdc-par
ameter laden":printtab(20)c4$"
6) vdc-parameter speichern":pr
inttab(20)c4$"7) vdc mit einge
gebenen parametern fuellen"    <jc>
150 printtab(20)c4$"8) paramet
er in vdc speichern : ";:ifzx=
0thenprint"sofort":goto170     <eh>
160 print"erst nach schritt '7
)'"                            <lk>
170 printtab(20)c4$"9) ende"   <ed>
180 printtab(20)c4$c4$"testbil
d mit taste 'space'"           <hp>
190 getkeya$:ifa$="9"thenprint
cl$:clr:end                    <bm>
200 ifa$="1"then310            <mm>
210 ifa$="2"then510            <mf>
220 ifa$="3"then410            <da>
230 ifa$="4"then460            <cl>
240 ifa$="5"then600            <bg>
250 ifa$="6"then680            <kb>
260 ifa$="7"thenprintcl$+"regi
ster : ":fori=0to36:printhe$ta
b(12)"     "left$(ql$,4)i:sysde
c("cdcc"),in(i),i:next:goto120 <cf>
270 ifa$="8"thenzx=not(zx):got
o120                           <ap>
280 ifa$="9"then510            <en>
290 ifa$=" "then390            <he>
300 goto190                    <ie>
310 printcl$"nummer des regist
ers ";:inputnu:ifnu<0ornu>36th
en310                          <dl>
320 printc4$c4$"inhalt :"      <kj>
```

```
330 printc4$c4$"-normalerweise
 : +";a(nu);" = $";right$(hex$
(a(nu)),2);" = %";:c=a(nu):gos
ub1260:printc$                 <gk>
340 printc4$"-aktueller inhalt
 : ";:x=in(nu):print"+";x;" =
$";right$(hex$(x),2);" = %";:c
=x:gosub1260:printc$           <od>
350 printc4$c4$c4$"bemerkung :
":printc4$ax$(nu)              <dn>
360 gosub1110                  <nc>
370 in(nu)=c:ifzx=0thensysdec(
"cdcc"),c,nu                   <ac>
380 goto120                    <hf>
390 printre$"a"gr$"a"bl$"a";:g
eta$:ifa$=""then390            <lp>
400 goto120                    <lo>
410 printcl$"nummer des zu les
enden registers ";:inputnu:ifn
u<0ornu>36then410              <eh>
420 printc4$c4$"inhalt :"      <jl>
430 printc4$c4$"-normalerweise
 : +";a(nu);" = $";right$(hex$
(a(nu)),2);" = %";:c=a(nu):gos
ub1260:printc$                 <mf>
440 printc4$"-aktueller inhalt
 : +";:x=in(nu):printx;" = $";
right$(hex$(x),2);" = %";:c=x:
gosub1260:printc$              <bn>
450 printc4$c4$c4$"bemerkung :
":printc4$ax$(nu):getkeya$:got
o120                           <lo>
460 nu=0                       <de>
470 printcl$"register-nummer :
";nu;"="a(nu):sysdec("cdcc"),
a(nu),nu                       <pd>
480 printc4$"mit 'space' naech
stes register zuruecksetzen"   <oh>
490 getkeya$:ifa$=" "andnu<36t
hennu=nu+1:goto470             <ff>
500 goto120                    <pi>
510 printcl$"nummer des regist
ers ";:inputnu:ifnu<0ornu>36th
en510                          <hn>
520 printc4$c4$"inhalt :"      <ff>
530 printc4$c4$"-normalerweise
 : ";a(nu):printc4$"-aktueller
 inhalt : ";:x=in(nu):printx:p
rintc4$c4$c4$"bemerkung :":pri
ntc4$ax$(nu)                   <ab>
540 getkeya$:ifa$=chr$(13)then
120                            <ed>
550 ifa$="+"andx<255thenx=x+1:
printchr$(7);:goto580          <gg>
560 ifa$<>"-"orx=0then540      <aj>
570 x=x-1:printchr$(7);        <ac>
580 in(nu)=x:ifzx=0thensysdec(
"cdcc"),x,nu                   <ag>
590 printhe$left$(qd$,8)tab(20
)"    "left$(ql$,4);x:goto540  <ef>
600 printcl$"
                  vdc-daten laden
"                              <gm>
610 print"
            ---------------"c4
$c4$                           <jm>
620 print"dateiname ($=directo
ry) ";na$:ifleft$(na$,1)="$"th
enprintcl$;:directory:getkeya$
:goto600                       <an>
630 printc4$" sind sie sicher
?":getkeya$:ifa$="n"then120    <ji>
640 open1,8,0,na$+",s":ifds<>0
thenclose1:printc4$c4$re$"file
 not found !":getkeya$:printwh
$c2$"                 ":printle
ft$(qu$,4)"                   "
c2$c2$c2$:goto620              <ck>
650 printc4$c4$"ich lese :"c4$ <ma>
660 print"register nr.:":fori=
0to36:input#1,in(i):printc2$ta
b(13)i:ifzx=0thensysdec("cdcc"
),in(i),i                      <ac>
670 next:close1:goto120        <ch>
680 printcl$"
              vdc-daten speiche
rn"                            <pj>
690 print"
         -------------------"
c4$c4$                         <ad>
700 print"dateiname ($=directo
ry) ";na$:ifleft$(na$,1)="$"th
enprintcl$;:directory:getkeya$
:goto680                       <mo>
710 open1,8,4,"@0:"+na$+",s,w" <jb>
720 printc4$c4$"ich schreibe :
"c4$                           <hp>
730 print"register nr.:":fori=
0to36:print#1,in(i):printc2$ta
b(13)i:next:close1:goto120     <nf>
740 data 126,"totalanzahl der
zeichen/zeile + strahlenrueckl
auf"                           <bh>
750 data 080,"anzahl der zeich
en/zeile"                      <kn>
760 data 102,"synchronisation
des linken randes"             <hg>
770 data 073,"bit 0-3 : horizo
ntale synch.-puls-breite / bit
 4-7 : vertikale"              <am>
780 data 039,"totalanzahl der
zeilen + strahlenruecklauf"    <fb>
790 data 224,"bit 0-4 : feinei
nstellung von reg.4 / bit 5-7
: immer gesetzt"               <ee>
800 data 025,"anzahl der zeile
```

```
n"                                      <nd>
810 data 032,"oberer rand des
bildschirms"                            <ep>
820 data 252,"bit 0-1 : 00/10=
non-interlace-mode : 01/11=int
erlace-mode"                            <mi>
830 data 231,"bit 0-4 : anzahl
 der rasterzeilen/zeichen-1  b
it 5-7 = 1"                             <co>
840 data 160,"bit 5-6 : cursor
-mode"                                  <ol>
850 data 231,"bit 0-4 : end-ze
ile des cursors"                        <cn>
860 data 000,"hi-byte des vide
o-rams"                                 <bp>
870 data 000,"lo-byte des vide
o-rams (mit reg.12)"                    <ek>
880 data 000,"hi-byte der curs
orposition"                             <lj>
890 data 000,"lo-byte der curs
orposition"                             <no>
900 data 000,"light-pen-positi
on vertikal"                            <el>
910 data 000,"light-pen-positi
on horizontal"                          <ei>
920 data 000,"hi-byte der zu a
endernden adresse"                      <fd>
930 data 000,"lo-byte der zu a
endernden adresse"                      <pc>
940 data 008,"hi-byte von attr
ibut-ram"                               <ie>
950 data 000,"lo-byte von attr
ibut-ram"                               <ik>
960 data 120,"bit 0-3 : breite
 eines zeichens  bit 4-7 :tota
lanzahl der zeilen"                     <lm>
970 data 232,"laenge eines zei
chens"                                  <dd>
980 data 032,"bit7:copy-bit bi
t6:rvs-bit bit0-4:vertikale ra
nd in rasterzeilen"                     <be>
990 data 071,"bit7:text-bit bi
t6:atr-bit bit5:semi-grafik bi
t4:breite bit0-3:horiz.rand"            <lj>
1000 data 240,"bit 0-3 : hinte
rgrundfarbe  bit 4-7 : vorderg
rundfarbe"                              <hl>
1010 data 000,"spaltenaufteilu
ng"                                     <bn>
1020 data 047,"bit 5-7 : zeich
engenerator  bit 4 : ram numme
r"                                      <lj>
1030 data 231,"bit 0-4 : zeile
 zum unterstreichen"                    <de>
1040 data 001,"zeichen-anzahl
zum schreiben/kopieren"                 <ei>
1050 data 032,"daten zum schre
iben"                                   <fg>
1060 data 000,"startadresse hi
 zum kopieren"                          <kk>
1070 data 000,"startadresse lo
 zum kopieren"                          <fg>
1080 data 125,"zeichen von beg
inn bis pos. display-enable-pi
ns"                                     <mh>
1090 data 100,"zeichen von beg
inn bis neg. display-enable-pi
ns"                                     <ea>
1100 data 245,"bit 0-3 : refre
sh-rate"                                <eh>
1110 printc4$"eingabe "rn$"h"r
f$"ex,"rn$"d"rf$"ez oder "rn$"
b"rf$"inaer ?"c4$:getkeya$              <ai>
1120 ifa$="d"theninput"dezimal
-wert ";c:ifc<0orc>255thenprin
tc2$;:goto1120                          <oj>
1130 ifa$="d"thengosub1260:got
o1190                                   <dg>
1140 ifa$="h"theninput"hexadez
imal-wert ";c$:c=dec(c$):ifc<0
orc>255thenprintc2$;:goto1140           <bj>
1150 ifa$="h"thengosub1260:got
o1190                                   <na>
1160 ifa$<>"b"thenprintc2$c2$c
2$;:goto1110                            <ag>
1170 input"binaer-wert (8 stel
len) ";c$:iflen(c$)<>8thenprin
tc2$;:goto1170                          <ia>
1180 gosub1240:gosub1260                <pa>
1190 printc2$c2$c2$"

        ":printc4$"
                              "         <af>
1200 printc4$"

                       "                <am>
1210 printleft$(qu$,4)" +";c;"
 = $";right$(hex$(c),2);" = %"
;c$                                     <ap>
1220 printc4$c4$"eingabe richt
ig ?":getkeya$:ifa$="j"thenret
urn                                     <on>
1230 printleft$(qu$,4)"

":printc4$c4$"
  "left$(qu$,6):goto1110                <mf>
1240 c=0:fori=1to8:n$=mid$(c$,
i,1):ifn$="1"thenc=c+2^(8-i)            <lk>
1250 nexti:return                       <ka>
1260 c$="":i1=c:fori=1to8:ifi1
>=2^(8-i)thenc$=c$+"1":i1=i1-2
^(8-i):next:return                      <fc>
1270 c$=c$+"0":next:return              <pf>
60000 rem nachspann =========           <dg>
60010 rem farb-/steuercodes             <ij>
60020 wh$=chr$(005):c4$=chr$(0
```

VDC — MON — 4 —

```
17)                                <ee>
60030 rn$=chr$(018):he$=chr$(0
19)                                <nh>
60040 re$=chr$(028):gr$=chr$(0
30)                                <ah>
60050 bl$=chr$(031):c2$=chr$(1
45)                                <bo>
60060 rf$=chr$(146):cl$=chr$(1
47)                                <oj>
60070 c1$=chr$(157)                <ih>
60080 rem zeichenfolgen            <fe>
60090 for q=1 to 40                <ka>
60100 qd$=qd$+c4$:qu$=qu$+c2$      <bi>
60110 ql$=ql$+c1$                  <jj>
60120 next q                       <fp>
60130 return                       <ch>
```

HOCHAUFLÖSENDE GRAFIK IM VDC

Die Möglichkeit, mit dem C 128 achtzig Zeichen auf den Bildschirm zu bringen, ist schon eine feine Sache. Interessant wird's aber erst, wenn Sie diesen in hochauflösenden Grafikmodus schalten. Wie's funktioniert, zeigt der folgende Beitrag.

So interessant der 80-Zeichen-Bildschirm für professionelle Anwendungen im C128 im Textmodus sein kann, für unsere Freude der Grafik ist die Tatsache entscheidend, den VDC-Chip auch in den hochauflösenden Modus versetzen zu können. Damit ist es möglich, insgesamt 128.000 Bildschirm-Pixel einzeln oder zu einem Byte (= 8 Pixel) zusammengefaßt, anzusprechen.

Zuständig unter den 37 VDC-Registern ist dafür Nr. 25, das sich um die Betriebsart des 80-Zeichen-Monitors kümmert. Ist das Bit 7 (Wert: 128) gelöscht, so ist immer der Textmodus vorherrschend. Ist es gesetzt, so wird der Hires-Grafikmodus eingeschaltet. Ab sofort stehen dann in horizontaler Richtung 640 Bildpunkte, in vertikaler Richtung nach wie vor 200 Pixel zur Verfügung.

Da die Byte zu je acht Pixel innerhalb einer Bildschirmzeile hintereinander abgelegt werden (also so, wie sie kommen), ist die Programmierung einer solchen VDC-Bitmap doch relativ einfach. Was ein wenig stört: Die bekannten Grafikbefehle des BASIC 7.0 im C128 wie DRAW, CIRCLE, BOX haben beim VDC keine Chance, er ignoriert sie einfach.

Am bestehn übergeben Sie einem selbstgefertigten Programm gewisse Koordinatenpunkte auf dem Hires-Bildschirm und zeichnen auf diese Art und Weise ein Bild in die VDC-Bitmap. Die grundlegenden Programmierschritte zeigt Ihnen das folgende kleine Listing.

hb ■

VDC — Hires — 1 —

```
10 rem hochaufloesende ====128 <ic>
11 rem grafik im vdc       --- <ld>
20 rem (c) commodore welt/h.b. <dp>
30 rem ======================= <ae>
40 rem version 7.0 80z/ascii== <fi>
50 rem 128 pc + kass./floppy== <bk>
60 rem ======================= <ee>
70 scnclr:fast:rem auf 2 hz-ta
kt schalten                    <ab>
80 cr=54784:rem variable fuer
vdc-kommunikations-adresse 1   <oo>
90 ci=54785:rem  ''
      ''          ''     2     <nm>
100 ba=1: rem beginn bitmap    <jb>
110 p=20:rem in die bitmap-adr
esse einzutragender byte-wert  <ii>
120 bank15: rem entspr. bank e
inschalten                     <on>
130 pokecr,25: rem reg.25 =zus
taendig fuer vdc-betriebsart   <jh>
140 pokeci,128: rem wert fuer
reg. 25                        <ep>
150 pokecr,26: rem reg.26 = fa
rbgebung im vdc (vorder/hinter
grund)                         <of>
160 pokeci,10: rem wert fuer r
eg. 26                         <fj>
170 fori=0to16383:rem einleses
chleife zum beschreiben der bi
tmap                           <ln>
180 gosub220                   <ba>
190 next                       <pp>
200 goto200                    <cj>
210 rem ** upr zum beschreiben
 der vdc-bitmap **             <ib>
220 hb=int(ba/256):lb=ba-hb*25
6:rem berechnen der bildschirm
adresse                        <mh>
230 pokecr,18:pokeci,hb:rem hi
byte bitmap-adresse            <bl>
240 pokecr,19:pokeci,lb:rem lo
byte bitmap-adresse            <hn>
250 pokecr,31:pokeci,p:rem reg
.schreiben in bitmap-adresse   <na>
260 pokecr,30:pokeci,16:rem re
g.wie oft 'p' geschrieben wird <eh>
270 ba=ba+1:rem erhoehen der b
itmap-adresse                  <ej>
280 return                     <ma>
```

SCRIPT CALL KOMFORTABEL TEXTVERARBEITUNG FÜR DEN C128

Gutes muss nicht teuer sein!

Dieses Programm nutzt die Fähigkeiten Ihres C128 im 80-Zeichen-Modus mit entsprechendem Bildschirm voll aus.

Für ein gutes Textverarbeitungs-Programm sind mittlerweile gewisse Standardfunktionen wie Blockoperationen, automatischer Zeilenumbruch (Word-Wrapping), Suchfunktion und informative Hilfsseite selbstverständlich geworden. All diese Möglichkeiten besitzt Scriptcall, also kann es zu den guten gerechnet werden.
Es arbeitet nach dem Prinzip WYSIWYG (What you see, is what you get), das heißt, der Text ist so auf dem Bildschirm zu sehen, wie er später zu Papier gebracht wird. Zudem zeichnet sich das Programm durch ansprechende Menüsteuerung unter Benutzung mehrerer Windows sowie seine recht einfache Handhabung aus – eine echte Alternative zu vielen teueren kommerziellen Textverarbeitungs-Programmen für den C128.

TIPP-MARATHON

Allerdings: Vor den Lohn haben die Götter den Schweiß gesetzt. Eine recht umfangreiche Eintipparbeit erwartet Sie.
Verwenden Sie dazu unbedingt unseren CHECKMON, den Checksummer für Maschinensprache-Listings.
Entsprechende Bedienungshinweise finden Sie bei der Beschreibung zu dieser Eingabehilfe.

ANWEISUNGEN ZUM ABSPEICHERN DER EINZELNEN FILES

Das richtige Speichern der im MONITOR-Modus abgetippten einzelnen Programmteile ist mindestens ebenso wichtig wie das Abtippen selbst, wenn das Programm anschließend exakt laufen soll.
Hier eine Übersicht der Speicheranweisungen aus dem Maschinensprache-MONITOR des C128:

SCRIPTCALL (Hauptprogramm)
S "SCRIPT-CALL" 08 1C01 57D1
SC1
S "SC 1" 08 0F78 0FFF
!C 2
S "SC 2" 08 1300 1597
SC 3
S "SC 3" 08 1300 1BB6
=C 4
S "SC 4" 08 0B00 0EF7
=C 5
S "SC 5" 08 1300 179D
SC 6
S "SC 6" 08 1300 1C51
SC 7
S "SC 7" 08 3620 363B

Achten Sie unbedingt auf die exakte Schreibweise dieser Monitor-Anweisungen!
Zum Schluß gibt's noch ein kleines BASIC-Programm, das zwar mit dem Hauptprogramm nicht direkt etwas zu tun hat, Ihnen aber wertvolle Hilfe zur Anpassung Ihres Druckers an dieses Text-Programm leisten kann. Sie tippen es mit Hilfe unseres Checksummers ab.

EIGENE PROGRAMM- UND ARBEITSDISKETTE ANLEGEN

Haben Sie die Tipparbeit unbeschadet hinter sich gebracht, so müßten folgende Files auf Ihrer Programmdiskette zu finden sein:
SCRIPT CALL
SC 1
SC 2
SC 3
SC 4
SC 5
SC 6
SC 7
INSTALL EPSON

UM MIT SCRIPTCALL ZU ARBEITEN, GEHEN SIE FOLGENDERMASSEN VOR:

Laden Sie das Programm mit DLOAD"SCRIPT-CALL" und starten es mit RUN,
oder geben Sie gleich RUN"SCRIPTCALL" ein,
oder einfach SHIFT/RUN-STOP drücken.
Nach dem Start des Programmes ist im Titelfenster die gewünschte Textbreite einzustellen (Cursor nach links = schmäler, Cursor rechts = breiter).
Die oberste Bildschirmzeile enthält Informationen über die aktuelle Cursorposition (Zeile/Spalte), die eingestellte Seite, die Art der Textdarstellung (Revers/Unterstrichen), den Programm-Modus (write/command-Funktion) sowie den Schreib-Modus (overwrite/insert). Darunter befindet sich die Anzeige der Tabulatoren und in der untersten Bildschirmzeile das Hauptmenü. Die drei unterschiedlichen Programm-Modi seien im folgenden erklärt:

WRITE-Modus:
Dies ist der Schreib-Modus, in dem man sich zunächst befindet. Hier können die Texte editiert und durch folgende Funktionen manipuliert werden:

HOME	– setzt den Cursor in die linke obere Ecke.
CLR	– setzt den Cursor an den Seitenanfang.
RETURN	– setzt den Cursor an den Anfang der nächsten Zeile.
INST	– fügt an der Cursorposition ein Leerzeichen ein.
DEL	– löscht das Zeichen links vom Cursor. – wird DEL in der ersten Spalte gedrückt, so wird die darüberliegende Zeile mit dem Text aus dieser Zeile aufgefüllt. Das Gegenstück hierzu ist:
SHFT/RET	– trennt diese Zeile an der Cursorposition und schiebt den Resttext dieser Zeile in die nächste.

Bitte lesen Sie weiter auf Seite 82

```
monitor
    pc  sr ac xr yr sp

; fb000 00 00 00 00 f8

>01c01 1e 1c 01 00 8f 20 2a 2a:<f4>
>01c09 2a 2a 2a 2a 2a 2a 2a 2a:<2c>
>01c11 2a 2a 2a 2a 2a 2a 2a 2a:<1d>
>01c19 2a 2a 2a 2a 00 3b 1c 02:<34>
>01c21 00 8f 20 2a 2a 2a 2a 2a:<b4>
>01c29 20 53 43 52 49 50 54 43:<99>
>01c31 41 4c 4c 20 2a 2a 2a 2a:<6f>
>01c39 2a 00 58 1c 03 00 8f 20:<da>
>01c41 2a 2a 2a 2a 2a 2a 2a 2a:<7c>
>01c49 2a 2a 2a 2a 2a 2a 2a 2a:<aa>
>01c51 2a 2a 2a 2a 2a 2a 00 75:<c9>
>01c59 1c 04 00 8f 20 2a 20 28:<d2>
>01c61 50 29 20 43 4f 4d 4d 4f:<0c>
>01c69 44 4f 52 45 20 44 49 53:<47>
>01c71 43 20 2a 00 92 1c 05 00:<d8>
>01c79 8f 20 2a 20 28 43 29 20:<cc>
>01c81 4d 2e 20 53 43 48 4f 4c:<f2>
>01c89 5a 20 20 20 20 20 20 2a:<6c>
>01c91 00 af 1c 06 00 8f 20 2a:<2c>
>01c99 20 20 20 20 20 20 20 20:<4f>
>01ca1 20 20 20 20 20 20 20 20:<5f>
>01ca9 20 20 20 20 2a 00 cc 1c:<f4>
>01cb1 07 00 8f 20 2a 20 56 20:<91>
>01cb9 37 2e 30 20 20 20 20 38:<0e>
>01cc1 30 5a 20 2f 20 44 49 4e:<60>
>01cc9 20 2a 00 e9 1c 08 00 8f:<f5>
>01cd1 20 2a 20 31 32 38 2f 46:<57>
>01cd9 4c 4f 50 50 59 2f 44 52:<df>
>01ce1 55 43 4b 45 52 20 2a 00:<b6>
>01ce9 06 1d 09 00 8f 20 2a 2a:<70>
>01cf1 2a 2a 2a 2a 2a 2a 2a 2a:<57>
>01cf9 2a 2a 2a 2a 2a 2a 2a 2a:<0a>
>01d01 2a 2a 2a 2a 00 1b 1d 0a:<3f>
>01d09 00 97 34 37 2c 34 38 3a:<3d>
>01d11 97 34 38 2c 31 38 32 3a:<34>
>01d19 9c 00 2d 1d 0b 00 fe 25:<4b>
>01d21 3a 99 22 13 13 22 3a de:<be>
>01d29 35 2c 31 00 64 1d 0c 00:<ae>
>01d31 43 24 b2 c7 28 32 37 29:<b6>
>01d39 3a 99 c7 28 31 34 29 c7:<fa>
>01d41 28 31 31 29 43 24 22 4e:<a0>
>01d49 22 43 24 22 5a 22 43 24:<c6>
>01d51 22 55 22 3a 41 50 b2 31:<f9>
>01d59 3a 46 50 b2 31 3a 4c 50:<b8>
>01d61 b2 35 00 bf 1d 0d 00 46:<06>
>01d69 31 24 b2 22 ce ce ce ce:<32>
>01d71 ce ce ce ce ce ce ce ce:<08>
>01d79 ce ce ce ce ce ce ce ce:<99>
>01d81 ce ce ce ce ce ce ce ce:<68>
>01d89 ce ce ce ce ce ce ce ce:<38>
>01d91 ce ce ce ce ce ce ce ce:<48>
>01d99 ce ce ce ce ce ce ce ce:<58>
>01da1 ce ce ce ce ce ce ce ce:<a4>
>01da9 ce ce ce ce ce ce ce ce:<78>
>01db1 ce ce ce ce ce ce ce ce:<05>
>01db9 ce ce ce ce 22 00 1a 1e:<5f>
>01dc1 0e 00 46 32 24 b2 22 5f:<f9>
>01dc9 5f 5f 5f 5f 5f 5f 5f 5f:<90>
>01dd1 5f 5f 5f 5f 5f 5f 5f 5f:<80>
>01dd9 5f 5f 5f 5f 5f 5f 5f 5f:<2f>
>01de1 5f 5f 5f 5f 5f 5f 5f 5f:<60>
>01de9 5f 5f 5f 5f 5f 5f 5f 5f:<4f>
>01df1 5f 5f 5f 5f 5f 5f 5f 5f:<bd>
>01df9 5f 5f 5f 5f 5f 5f 5f 5f:<ad>
>01e01 5f 5f 5f 5f 5f 5f 5f 5f:<1f>
>01e09 5f 5f 5f 5f 5f 5f 5f 5f:<0d>
>01e11 5f 5f 5f 5f 5f 5f 5f 22:<9f>
>01e19 00 68 1e 0f 00 46 33 24:<b1>
>01e21 b2 22 c9 11 9d c9 11 9d:<77>
>01e29 c9 11 9d c9 11 9d c9 11:<93>
>01e31 9d c9 11 9d c9 11 9d c9:<94>
>01e39 11 9d c9 11 9d c9 11 9d:<9e>
>01e41 c9 11 9d c9 11 9d c9 11:<a3>
>01e49 9d c9 11 9d c9 11 9d c9:<84>
>01e51 11 9d c9 11 9d c9 11 9d:<2c>
>01e59 c9 11 9d c9 11 9d c9 11:<ec>
>01e61 9d c9 11 9d c9 22 00 b6:<e4>
>01e69 1e 10 00 46 34 24 b2 22:<5e>
>01e71 ca 11 9d ca 11 9d ca 11:<8c>
>01e79 9d ca 11 9d ca 11 9d ca:<b5>
>01e81 11 9d ca 11 9d ca 11 9d:<31>
>01e89 ca 11 9d ca 11 9d ca 11:<95>
>01e91 9d ca 11 9d ca 11 9d ca:<65>
>01e99 11 9d ca 11 9d ca 11 9d:<02>
>01ea1 ca 11 9d ca 11 9d ca 11:<a5>
>01ea9 9d ca 11 9d ca 11 9d ca:<d6>
>01eb1 11 9d ca 22 00 11 1f 11:<78>
>01eb9 00 46 35 24 b2 22 20 20:<8b>
>01ec1 20 20 20 20 20 20 20 20:<99>
>01ec9 20 20 20 20 20 20 20 20:<a9>
>01ed1 20 20 20 20 20 20 20 20:<b9>
>01ed9 20 20 20 20 20 20 20 20:<c9>
>01ee1 20 20 20 20 20 20 20 20:<d9>
>01ee9 20 20 20 20 20 20 20 20:<69>
>01ef1 20 20 20 20 20 20 20 20:<f8>
>01ef9 20 20 20 20 20 20 20 20:<08>
>01f01 20 20 20 20 20 20 20 20:<18>
>01f09 20 20 20 20 20 20 22 00:<99>
>01f11 6c 1f 12 00 46 36 24 b2:<40>
>01f19 22 a6 a6 a6 a6 a6 a6 a6:<df>
>01f21 a6 a6 a6 a6 a6 a6 a6 a6:<73>
>01f29 a6 a6 a6 a6 a6 a6 a6 a6:<c2>
>01f31 a6 a6 a6 a6 a6 a6 a6 a6:<52>
>01f39 a6 a6 a6 a6 a6 a6 a6 a6:<22>
>01f41 a6 a6 a6 a6 a6 a6 a6 a6:<32>
>01f49 a6 a6 a6 a6 a6 a6 a6 a6:<02>
>01f51 a6 a6 a6 a6 a6 a6 a6 a6:<12>
>01f59 a6 a6 a6 a6 a6 a6 a6 a6:<e2>
>01f61 a6 a6 a6 a6 a6 a6 a6 a6:<f2>
>01f69 a6 22 00 ec 1f 13 00 45:<b8>
>01f71 53 24 b2 22 41 42 43 44:<b3>
>01f79 45 46 47 48 49 4a 4b 4c:<d0>
>01f81 4d 4e 4f 50 51 52 53 54:<15>
>01f89 55 56 57 58 59 5a c1 c2:<31>
>01f91 c3 c4 c5 c6 c7 c8 c9 ca:<5c>
>01f99 cb cc cd ce cf d0 d1 d2:<5e>
>01fa1 d3 d4 d5 d6 d7 d8 d9 da:<7c>
>01fa9 bc bb bd dc db dd be 3a:<13>
>01fb1 3b 3c 3d 3e 2f 3f 2b 2d:<4f>
>01fb9 31 32 33 34 35 36 37 38:<72>
```

```
>01fc1 39 30 21 b1 40 24 25 26:<0a>
>01fc9 2f 28 29 5b 5e 5c 5d a5:<a4>
>01fd1 a7 a3 a9 b0 a6 ff a8 aa:<dc>
>01fd9 df ab a2 a4 a1 ba b9 2c:<3a>
>01fe1 2e 20 5f 22 aa c7 28 33:<d2>
>01fe9 34 29 00 33 20 14 00 4c:<3d>
>01ff1 53 24 b2 22 41 42 43 44:<b4>
>01ff9 45 46 47 48 49 4a 4b 4c:<d5>
>02001 4d 4e 4f 50 51 52 53 54:<15>
>02009 55 56 57 58 59 5a bc bb:<97>
>02011 bd be 3b 3c 3d 3e 2f 2b:<40>
>02019 2d 31 32 33 34 35 36 37:<35>
>02021 38 39 30 21 24 25 26 2f:<9c>
>02029 28 29 5b 5e 5c 5d 2e 20:<d1>
>02031 22 00 6b 20 15 00 97 30:<5e>
>02039 2c c2 28 30 29 b0 36 34:<94>
>02041 3a 97 31 2c c2 28 31 29:<f3>
>02049 af 36 33 3a 97 34 31 30:<b7>
>02051 34 2c 30 3a 97 34 31 30:<2d>
>02059 35 2c 30 3a fe 0b 31 3a:<99>
>02061 97 32 37 35 37 2c 31 32:<72>
>02069 38 00 8d 20 16 00 e7 36:<17>
>02071 2c 37 3a e7 31 2c 31 3a:<ba>
>02079 de 31 2c 31 3a de 30 3a:<15>
>02081 de 35 3a 97 32 34 38 2c:<8e>
>02089 31 39 32 00 a2 20 17 00:<81>
>02091 d7 32 38 3a fe 15 3a 8b:<83>
>02099 44 53 b3 b1 30 a7 32 38:<0e>
>020a1 00 af 20 18 00 fe 11 22:<17>
>020a9 53 43 20 31 22 00 f2 20:<b3>
>020b1 19 00 fe 11 22 53 43 20:<7a>
>020b9 32 22 3a 9e 34 38 36 34:<b1>
>020c1 3a fe 11 22 53 43 20 33:<1d>
>020c9 22 3a fe 11 22 53 43 20:<26>
>020d1 34 22 3a 9e 33 36 33 37:<72>
>020d9 3a fe 11 22 53 43 20 35:<8f>
>020e1 22 3a 9e 33 36 36 31 3a:<99>
>020e9 fe 11 22 53 43 20 36 22:<8d>
>020f1 00 27 21 1a 00 9f 35 2c:<fd>
>020f9 38 2c 35 2c 22 53 43 20:<36>
>02101 37 2c 50 2c 52 22 3a 84:<8d>
>02109 35 2c 4c 43 2c 50 54 2c:<e3>
>02111 44 4e 2c 53 41 2c 4c 4d:<98>
>02119 2c 50 53 2c 4e 4c 3a fe:<c0>
>02121 0f 3a 89 33 30 00 34 21:<06>
>02129 1b 00 8f 20 46 45 48 4c:<92>
>02131 45 52 00 73 21 1c 00 99:<61>
>02139 22 13 13 93 22 3a 50 31:<9a>
>02141 b2 32 37 3a 50 32 b2 38:<34>
>02149 3a 50 33 b2 35 32 3a 50:<38>
>02151 34 b2 31 33 3a 55 24 b2:<eb>
>02159 22 d0 20 52 20 4f 20 42:<9d>
>02161 20 4c 20 45 20 4d 22 3a:<17>
>02169 8d 33 30 38 3a 8d 33 32:<fc>
>02171 33 00 99 21 1d 00 99 22:<ee>
>02179 11 20 c9 4e 53 45 52 54:<4e>
>02181 20 4f 52 49 47 49 4f 4e:<ea>
>02189 41 4c 20 44 49 53 4b 22:<69>
>02191 3a a1 f9 41 24 3a 8a 00:<cb>
>02199 b7 21 1e 00 d7 33 33 31:<ed>
>021a1 3a 81 54 b2 31 a4 38 3a:<b0>
>021a9 f9 54 2c c7 28 31 33 32:<f9>
>021b1 aa 54 29 3a 82 00 1b 22:<3d>
>021b9 1f 00 99 22 13 13 93 9f:<2f>
>021c1 12 20 20 3d 3d 92 30 30:<63>
>021c9 12 3d 3d 20 20 3d 3d 92:<35>
>021d1 30 30 12 3d 3d 20 20 3d:<f5>
>021d9 3d 92 30 31 12 3d 3d 20:<a7>
>021e1 20 20 20 2a 20 2a 20 2a:<3b>
>021e9 20 20 d3 43 52 49 50 54:<b6>
>021f1 43 41 4c 4c 20 20 2a 20:<d9>
>021f9 2a 20 2a 20 20 20 20 20:<44>
>02201 20 20 20 20 20 20 20 57:<bf>
>02209 52 49 54 45 20 4f 56 45:<b6>
>02211 52 57 52 49 54 45 20 91:<06>
>02219 22 00 3c 22 20 00 81 54:<b4>
>02221 b2 31 a4 38 31 a9 31 30:<25>
>02229 3a 99 22 18 20 20 20 20:<c6>
>02231 20 20 20 20 20 20 22 3b:<82>
>02239 3a 82 00 57 22 21 00 5a:<01>
>02241 43 b2 4c 43 3a 4c 43 b2:<ad>
>02249 37 39 3a 8d 33 30 31 3a:<29>
>02251 4c 43 b2 5a 43 00 b0 22:<5c>
>02259 22 00 99 22 12 20 20 d0:<3f>
>02261 41 47 45 20 20 20 20 20:<c4>
>02269 20 d3 45 41 52 43 48 20:<f8>
>02271 20 20 20 c2 4c 4f 43 4b:<ff>
>02279 20 20 20 20 20 c4 45 4c:<a7>
>02281 45 54 45 20 20 20 20 c4:<18>
>02289 49 53 4b 20 20 20 20 20:<6f>
>02291 20 d0 52 49 4e 54 20 20:<fd>
>02299 20 20 20 d3 45 54 55 50:<dc>
>022a1 20 20 20 20 20 d3 50 45:<d1>
>022a9 43 49 41 4c 20 22 00 b9:<53>
>022b1 22 23 00 8d 33 32 35 00:<18>
>022b9 03 23 24 00 50 31 b2 32:<ff>
>022c1 32 3a 50 32 b2 38 3a 50:<e3>
>022c9 33 b2 35 36 3a 50 34 b2:<46>
>022d1 31 36 3a 55 24 b2 22 d3:<4f>
>022d9 43 52 49 50 54 43 41 4c:<47>
>022e1 4c 20 28 43 29 20 31 39:<2b>
>022e9 38 37 20 42 59 20 c2 2d:<67>
>022f1 c3 20 d3 4f 46 54 22 3a:<66>
>022f9 8d 33 30 38 3a 8d 33 32:<96>
>02301 33 00 22 23 25 00 41 b2:<a5>
>02309 31 3a 41 24 28 31 29 b2:<bc>
>02311 22 c3 4f 4c 55 4d 4e 53:<74>
>02319 3a 20 22 3a 8d 33 31 32:<aa>
>02321 00 43 23 26 00 e0 2c 31:<09>
>02329 35 2c 33 2c 22 0f b2 b3:<32>
>02331 b4 b5 22 3a e0 2c 31 35:<71>
>02339 2c 34 2c 22 b6 b7 b8 ae:<73>
>02341 22 00 72 23 27 00 99 22:<4c>
>02349 1d 22 3a fe 1a 32 34 2c:<ea>
>02351 31 31 2c 35 35 2c 31 35:<71>
>02359 3a e0 2c 31 32 2c 31 2c:<04>
>02361 22 90 22 aa c9 28 c4 28:<95>
>02369 4c 43 aa 31 29 2c 32 29:<ad>
>02371 00 7b 23 28 00 a1 f9 41:<da>
>02379 24 00 99 23 29 00 8b 41:<8e>
>02381 24 b2 22 1d 22 af 4c 43:<c9>
>02389 b3 37 39 a7 4c 43 b2 4c:<2d>
>02391 43 aa 31 3a 89 33 39 00:<b2>
>02399 b7 23 2a 00 8b 41 24 b2:<9e>
>023a1 22 9d 22 af 4c 43 b1 31:<bb>
>023a9 39 a7 4c 43 b2 4c 43 ab:<f0>
```

```
>023b1 31 3a 89 33 39 00 c8 23:<fe>
>023b9 2b 00 8b 41 24 b2 c7 28:<34>
>023c1 31 33 29 a7 34 35 00 d1:<72>
>023c9 23 2c 00 89 20 34 30 00:<d2>
>023d1 07 24 2d 00 99 22 9f 13:<0a>
>023d9 13 22 3a 8d 33 30 31 3a:<e5>
>023e1 99 43 24 22 45 22 43 24:<d1>
>023e9 22 46 22 3a fe 1a 30 2c:<c4>
>023f1 32 2c 37 39 2c 32 32 2c:<ce>
>023f9 31 3a fe 1a 30 2c 32 2c:<9d>
>02401 4c 43 2c 32 32 00 20 24:<37>
>02409 2e 00 97 38 34 32 2c 31:<91>
>02411 39 3a 97 32 30 38 2c 31:<8f>
>02419 3a 9e 34 38 36 35 00 28:<7c>
>02421 24 2f 00 fe 09 41 00 34:<2b>
>02429 24 30 00 8b 41 b2 31 a7:<31>
>02431 38 30 00 57 24 31 00 99:<0e>
>02439 22 13 13 22 a3 36 32 29:<4e>
>02441 22 12 43 4f 4d 4d 41 4e:<46>
>02449 44 92 22 3b 3a 92 32 31:<69>
>02451 31 2c 34 2c 34 00 85 24:<37>
>02459 32 00 a1 58 24 3a 8b 28:<86>
>02461 c2 28 32 31 31 29 af 34:<09>
>02469 29 b2 34 a7 92 32 31 31:<41>
>02471 2c 34 2c 34 3a 89 37 38:<8b>
>02479 3a d5 8b 58 24 b2 22 22:<f8>
>02481 a7 35 30 00 94 24 33 00:<83>
>02489 8b 58 24 b2 22 39 22 a7:<b6>
>02491 fe 18 00 d5 24 34 00 97:<30>
>02499 34 38 36 34 2c ce 08 28:<04>
>024a1 c2 28 34 38 36 34 29 2c:<04>
>024a9 31 29 3a 8b 52 56 b2 31:<cc>
>024b1 a7 52 56 b2 30 3a e0 2c:<12>
>024b9 36 30 2c 30 2c 22 12 20:<fb>
>024c1 22 3a d5 e0 2c 36 30 2c:<fb>
>024c9 30 2c 22 12 52 22 3a 52:<78>
>024d1 56 b2 31 00 e0 24 35 00:<13>
>024d9 fe 19 3a 89 37 38 00 ef:<2f>
>024e1 24 36 00 8b 58 24 b2 22:<52>
>024e9 55 22 a7 fe 18 00 30 25:<bb>
>024f1 37 00 97 34 38 36 34 2c:<a7>
>024f9 ce 08 28 c2 28 34 38 36:<9d>
>02501 34 29 2c 32 29 3a 8b 55:<4f>
>02509 4e b2 31 a7 55 4e b2 30:<a6>
>02511 3a e0 2c 35 38 2c 30 2c:<b6>
>02519 22 12 20 22 3a d5 e0 2c:<7b>
>02521 35 38 2c 30 2c 22 12 55:<f4>
>02529 22 3a 55 4e b2 31 00 3b:<2a>
>02531 25 38 00 fe 19 3a 89 37:<a3>
>02539 38 00 4a 25 39 00 8b 58:<c0>
>02541 24 b2 22 41 22 a7 fe 18:<44>
>02549 00 9b 25 3a 00 97 33 39:<b1>
>02551 37 31 2c ce 08 28 c2 28:<8a>
>02559 33 39 37 31 29 2c 31 29:<cc>
>02561 3a 8b 49 53 b2 31 a7 49:<3e>
>02569 53 b2 30 3a e0 2c 37 30:<09>
>02571 2c 30 2c 22 12 4f 56 45:<76>
>02579 52 57 52 49 54 45 22 3a:<1e>
>02581 d5 e0 2c 37 30 2c 30 2c:<81>
>02589 22 12 20 20 20 49 4e 53:<58>
>02591 45 52 54 22 3a 49 53 b2:<fa>
>02599 31 00 a6 25 3b 00 fe 19:<0c>
>025a1 3a 89 37 38 00 d1 25 3c:<51>
>025a9 00 8b 58 24 b2 22 09 22:<3e>
>025b1 a7 59 53 b2 c2 28 33 39:<45>
>025b9 36 39 29 3a e0 2c 59 53:<3b>
>025c1 2c 30 2c 22 18 22 3a 8d:<f1>
>025c9 33 30 31 3a 89 37 38 00:<1b>
>025d1 f9 25 3d 00 8b 58 24 b2:<a6>
>025d9 22 50 22 a7 fe 1a 30 2c:<9c>
>025e1 32 2c 4c 43 2c 32 32 3a:<f0>
>025e9 9e 33 31 35 30 3a 99 22:<a9>
>025f1 13 13 22 3a 89 37 38 00:<17>
>025f9 21 26 3e 00 8b 58 24 b2:<84>
>02601 22 51 22 a7 fe 1a 30 2c:<af>
>02609 32 2c 4c 43 2c 32 32 3a:<c0>
>02611 9e 33 31 38 36 3a 99 22:<d5>
>02619 13 13 22 3a 89 37 38 00:<65>
>02621 39 26 3f 00 8b 58 24 b2:<90>
>02629 22 4a 22 a7 97 33 39 36:<ad>
>02631 39 2c 30 3a 89 37 38 00:<25>
>02639 96 26 40 00 8b 58 24 b2:<81>
>02641 22 4b 22 a7 fe 1a 30 2c:<22>
>02649 c2 28 33 39 36 38 29 2c:<0a>
>02651 37 39 2c c2 28 33 39 36:<81>
>02659 38 29 3a e0 2c c2 28 33:<e9>
>02661 39 36 39 29 2c 30 2c c7:<e3>
>02669 28 32 37 29 aa 22 4b 22:<dd>
>02671 3a 97 33 39 36 39 2c c2:<f4>
>02679 28 32 33 36 29 3a 97 33:<2a>
>02681 39 36 38 2c c2 28 32 33:<58>
>02689 35 29 3a 99 22 13 13 22:<c8>
>02691 3a 89 37 38 00 be 26 41:<14>
>02699 00 8b 58 24 b2 22 49 22:<a5>
>026a1 a7 fe 1a 30 2c 32 2c 4c:<8f>
>026a9 43 2c 32 32 3a 9e 32 38:<0e>
>026b1 31 36 3a 99 22 13 13 22:<0e>
>026b9 3a 89 37 38 00 e6 26 42:<aa>
>026c1 00 8b 58 24 b2 22 44 22:<82>
>026c9 a7 fe 1a 30 2c 32 2c 4c:<80>
>026d1 43 2c 32 32 3a 9e 32 39:<3b>
>026d9 37 35 3a 99 22 13 13 22:<81>
>026e1 3a 89 37 38 00 05 27 43:<5d>
>026e9 00 8b 58 24 b2 22 5a 22:<18>
>026f1 a7 99 c7 28 32 37 29 22:<a4>
>026f9 5a 22 3a 8d 33 30 31 3a:<6b>
>02701 89 37 38 00 41 27 44 00:<f7>
>02709 8b 58 24 b2 22 59 22 a7:<a0>
>02711 99 c7 28 32 37 29 22 5a:<c8>
>02719 22 3a 81 54 b2 31 a4 38:<c2>
>02721 31 a9 31 30 3a 99 22 18:<7d>
>02729 1d 1d 1d 1d 1d 1d 1d 1d:<f8>
>02731 1d 1d 22 3b 3a 82 3a 8d:<5c>
>02739 33 30 31 3a 89 37 38 00:<37>
>02741 69 27 45 00 8b 58 24 b2:<17>
>02749 22 4c 22 a7 fe 1a 30 2c:<77>
>02751 32 2c 4c 43 2c 32 32 3a:<50>
>02759 9e 33 32 32 32 3a 99 22:<d8>
>02761 13 13 22 3a 89 37 38 00:<f2>
>02769 91 27 46 00 8b 58 24 b2:<73>
>02771 22 52 22 a7 fe 1a 30 2c:<ce>
>02779 32 2c 4c 43 2c 32 32 3a:<61>
>02781 9e 33 32 33 39 3a 99 22:<a3>
>02789 13 13 22 3a 89 37 38 00:<02>
>02791 b9 27 47 00 8b 58 24 b2:<10>
>02799 22 4d 22 a7 fe 1a 30 2c:<19>
```

```
>027a1 32 2c 4c 43 2c 32 32 3a:<70>
>027a9 9e 33 35 30 39 3a 99 22:<a7>
>027b1 13 13 22 3a 89 37 38 00:<9b>
>027b9 1d 28 48 00 8b 58 24 b2:<1d>
>027c1 22 2b 22 a7 99 22 1d 22:<ce>
>027c9 3a fe 1a 30 2c 32 2c 4c:<14>
>027d1 43 2c 32 32 3a 9e 33 36:<fb>
>027d9 33 37 3a 9e 36 38 39 31:<61>
>027e1 2c 30 2c 33 32 2c 2c 2c:<0a>
>027e9 37 32 36 34 2c 31 31 38:<e1>
>027f1 32 34 3a 9e 36 38 39 31:<cb>
>027f9 2c 30 2c 30 2c 2c 2c 31:<a0>
>02801 31 38 32 34 2c 31 36 33:<42>
>02809 38 34 3a 9e 33 36 33 37:<12>
>02811 3a 99 22 93 13 13 22 3a:<0d>
>02819 89 37 38 00 83 28 49 00:<c2>
>02821 8b 58 24 b2 22 bd 22 a7:<37>
>02829 fe 18 3a 99 22 1d 22 3a:<21>
>02831 fe 1a 30 2c 32 2c 4c 43:<12>
>02839 2c 32 32 3a 4e b2 d1 28:<45>
>02841 c9 28 d2 28 c2 28 32 35:<98>
>02849 33 29 29 2c 32 29 aa c9:<4b>
>02851 28 d2 28 c2 28 32 35 32:<74>
>02859 29 29 2c 32 29 29 aa c2:<97>
>02861 28 33 39 36 39 29 3a 9e:<d5>
>02869 33 36 33 37 3a 9e 36 38:<c1>
>02871 39 31 2c 30 2c 33 32 2c:<9f>
>02879 2c 2c 4e 2c 31 31 38 32:<55>
>02881 34 00 bf 28 4a 00 9e 36:<f2>
>02889 38 39 31 2c 30 2c 30 2c:<94>
>02891 2c 2c 4e aa 35 37 ac 38:<97>
>02899 30 2c 31 36 33 38 34 3a:<74>
>028a1 9e 33 36 33 37 3a 99 22:<d2>
>028a9 1d 22 3a 9e 36 32 31 32:<47>
>028b1 3a 99 22 13 13 22 3a fe:<1a>
>028b9 19 3a 89 37 38 00 25 29:<66>
>028c1 4b 00 8b 58 24 b2 22 5d:<b3>
>028c9 22 a7 fe 18 3a 99 22 1d:<b3>
>028d1 22 3a fe 1a 30 2c 32 2c:<54>
>028d9 4c 43 2c 32 32 3a 4e b2:<73>
>028e1 d1 28 c9 28 d2 28 c2 28:<18>
>028e9 32 35 33 29 29 2c 32 29:<47>
>028f1 aa c9 28 d2 28 c2 28 32:<59>
>028f9 35 32 29 29 2c 32 29 29:<6a>
>02901 aa c2 28 33 39 36 39 29:<91>
>02909 3a 9e 33 36 33 37 3a 9e:<96>
>02911 36 38 39 31 2c 30 2c 33:<cd>
>02919 32 2c 2c 2c 37 32 36 34:<18>
>02921 2c 4e 3a 00 61 29 4c 00:<4b>
>02929 9e 36 38 39 31 2c 30 2c:<46>
>02931 30 2c 2c 2c 31 31 38 32:<84>
>02939 34 2c 4e aa 35 37 ac 38:<ef>
>02941 30 3a 9e 33 36 33 37 3a:<04>
>02949 99 22 1d 22 3a 9e 36 32:<3c>
>02951 31 32 3a 99 22 13 13 22:<67>
>02959 3a fe 19 3a 89 37 38 00:<d5>
>02961 69 29 4d 00 89 35 30 00:<13>
>02969 97 29 4e 00 99 22 13 22:<4d>
>02971 a3 36 31 29 22 12 20 20:<9c>
>02979 20 57 52 49 54 45 22 3a:<c6>
>02981 fe 1a 30 2c 32 2c 4c 43:<dd>
>02989 2c 32 32 3a 9e 34 38 38:<e9>
>02991 31 3a 89 34 37 00 a7 29:<65>
>02999 4f 00 8f 20 46 55 4e 4b:<a2>
>029a1 54 49 4f 4e 53 00 c4 29:<75>
>029a9 50 00 5a b2 30 3a 99 22:<00>
>029b1 13 13 22 a3 36 31 29 22:<40>
>029b9 12 46 55 4e 4b 54 49 4f:<12>
>029c1 4e 22 00 e8 29 51 00 e0:<0a>
>029c9 2c 5a ac 31 30 aa 31 2c:<6a>
>029d1 32 34 2c c7 28 31 34 32:<1c>
>029d9 29 aa 22 12 ac 92 9d 22:<97>
>029e1 aa c7 28 31 34 29 00 09:<39>
>029e9 2a 52 00 eb 3a a1 41 24:<ba>
>029f1 3a 41 b2 c2 28 32 31 32:<cf>
>029f9 29 3a ec fc 41 24 b3 b1:<65>
>02a01 22 22 b0 41 b2 36 34 00:<4c>
>02a09 35 2a 53 00 8b 41 24 b2:<b1>
>02a11 22 1d 22 af 5a b3 37 a7:<cb>
>02a19 e0 2c 5a ac 31 30 aa 31:<5f>
>02a21 2c 32 34 2c 22 20 22 2c:<63>
>02a29 31 3a 5a b2 5a aa 31 3a:<f4>
>02a31 89 38 31 00 61 2a 54 00:<b9>
>02a39 8b 41 24 b2 22 9d 22 af:<de>
>02a41 5a b1 30 a7 e0 2c 5a ac:<4f>
>02a49 31 30 aa 31 2c 32 34 2c:<8a>
>02a51 22 20 22 2c 31 3a 5a b2:<be>
>02a59 5a ab 31 3a 89 38 31 00:<0e>
>02a61 82 2a 55 00 8b 41 24 b2:<d7>
>02a69 43 24 a7 e0 2c 5a ac 31:<36>
>02a71 30 aa 31 2c 32 34 2c 22:<6b>
>02a79 20 22 2c 31 3a 89 37 38:<39>
>02a81 00 93 2a 56 00 8b 41 24:<21>
>02a89 b2 c7 28 31 33 29 a7 38:<c5>
>02a91 39 00 a6 2a 57 00 8b 41:<26>
>02a99 b2 36 34 a7 8d 33 33 37:<d9>
>02aa1 3a 89 38 31 00 ae 2a 58:<2e>
>02aa9 00 89 38 32 00 e8 2a 59:<b4>
>02ab1 00 e0 2c 5a ac 31 30 aa:<1b>
>02ab9 31 2c 32 34 2c 22 20 22:<ad>
>02ac1 2c 31 3a 91 5a aa 31 89:<9d>
>02ac9 39 31 2c 31 30 31 2c 31:<60>
>02ad1 31 35 2c 31 33 34 2c 32:<7f>
>02ad9 30 32 2c 31 34 37 2c 32:<ba>
>02ae1 36 31 2c 31 32 35 00 fb:<07>
>02ae9 2a 5a 00 8f 20 2a 2a 2a:<97>
>02af1 20 50 41 47 45 20 2a 2a:<00>
>02af9 2a 00 03 2b 5b 00 5a b2:<01>
>02b01 31 00 30 2b 5c 00 50 31:<e2>
>02b09 b2 33 31 3a 50 32 b2 34:<b4>
>02b11 3a 50 33 b2 34 39 3a 50:<5f>
>02b19 34 b2 31 39 3a 55 24 b2:<93>
>02b21 22 d0 20 41 20 47 20 45:<ae>
>02b29 22 3a 8d 33 30 38 00 46:<8a>
>02b31 2b 5d 00 99 22 1d 22 3a:<a9>
>02b39 fe 1a 33 32 2c 37 2c 34:<bc>
>02b41 38 2c 31 38 00 ad 2b 5e:<72>
>02b49 00 41 b2 36 3a 41 24 28:<7d>
>02b51 31 29 b2 22 d0 41 47 45:<7e>
>02b59 20 31 22 3a 41 24 28 32:<34>
>02b61 29 b2 22 d0 41 47 45 20:<9f>
>02b69 32 22 3a 41 24 28 33 29:<67>
>02b71 b2 22 d0 41 47 45 20 33:<0b>
>02b79 22 3a 41 24 28 34 29 b2:<44>
>02b81 22 d0 41 47 45 20 34 22:<65>
>02b89 3a 41 24 28 35 29 b2 22:<e2>
```

```
>02b91 d0 41 47 45 20 35 22 3a:<34>
>02b99 41 24 28 36 29 b2 22 d2:<74>
>02ba1 45 54 55 52 4e 22 3a 8d:<fd>
>02ba9 33 31 32 00 ba 2b 5f 00:<73>
>02bb1 41 b2 36 3a 8d 33 31 35:<f3>
>02bb9 00 f7 2b 60 00 8b 5a b2:<55>
>02bc1 36 a7 99 22 1d 22 3a fe:<28>
>02bc9 1a 30 2c 32 2c 37 39 2c:<89>
>02bd1 32 32 3a 99 22 1d 22 3a:<84>
>02bd9 9e 36 32 31 32 3a 99 22:<cd>
>02be1 13 13 22 43 24 22 45 22:<6b>
>02be9 43 24 22 46 22 3a 5a b2:<15>
>02bf1 30 3a 89 38 31 00 29 2c:<f9>
>02bf9 61 00 9e 33 36 33 37 3a:<f7>
>02c01 9e 36 39 37 33 2c 30 2c:<b0>
>02c09 31 2c 2c 2c 37 32 36 34:<31>
>02c11 2c 31 36 33 38 33 2c 28:<4e>
>02c19 41 50 ab 31 29 ac 39 31:<41>
>02c21 32 30 aa 31 30 32 34 00:<67>
>02c29 86 2c 62 00 9e 36 39 37:<f0>
>02c31 33 2c 31 2c 30 2c 2c 2c:<e9>
>02c39 28 5a ab 31 29 ac 39 31:<7c>
>02c41 32 30 aa 31 30 32 34 2c:<e1>
>02c49 28 5a ab 31 29 ac 39 31:<de>
>02c51 32 30 aa 31 30 32 34 aa:<bc>
>02c59 39 31 31 39 2c 37 32 36:<11>
>02c61 34 3a 9e 33 36 33 37 3a:<cf>
>02c69 41 50 b2 5a 3a 99 22 13:<ba>
>02c71 13 22 3a e0 2c 32 31 2c:<b8>
>02c79 30 2c c9 28 c4 28 41 50:<bc>
>02c81 29 2c 31 29 00 c7 2c 63:<54>
>02c89 00 97 32 35 30 2c 39 36:<c0>
>02c91 3a 97 32 35 31 2c 32 38:<5a>
>02c99 3a 97 32 35 32 2c 39 36:<fa>
>02ca1 3a 97 32 35 33 2c 32 38:<1b>
>02ca9 3a 97 33 39 36 38 2c 32:<89>
>02cb1 3a 97 33 39 36 39 2c 30:<3b>
>02cb9 3a 5a b2 36 3a 8d 33 32:<05>
>02cc1 39 3a 89 39 36 00 dc 2c:<e9>
>02cc9 64 00 8f 20 2a 2a 2a 20:<d1>
>02cd1 53 45 41 52 43 48 20 2a:<9d>
>02cd9 2a 2a 00 0d 2d 65 00 50:<ed>
>02ce1 31 b2 30 3a 50 32 b2 31:<20>
>02ce9 31 3a 50 33 b2 37 38 3a:<3b>
>02cf1 50 34 b2 31 36 3a 55 24:<19>
>02cf9 b2 22 d3 20 45 20 41 20:<f4>
>02d01 52 20 43 20 48 22 3a 8d:<ad>
>02d09 33 30 38 00 23 2d 66 00:<d4>
>02d11 99 22 1d 22 3a fe 1a 31:<24>
>02d19 2c 31 35 2c 37 37 2c 31:<e1>
>02d21 35 00 2d 2d 67 00 99 53:<b8>
>02d29 45 24 3b 00 44 2d 68 00:<a9>
>02d31 a1 f9 41 24 3a 8b 41 24:<97>
>02d39 b2 c7 28 31 33 29 a7 31:<a1>
>02d41 30 39 00 7a 2d 69 00 8b:<c3>
>02d49 41 24 b2 c7 28 32 30 29:<b7>
>02d51 a7 8b c3 28 53 45 24 29:<57>
>02d59 b1 30 a7 53 45 24 b2 c8:<8f>
>02d61 28 53 45 24 2c c3 28 53:<5f>
>02d69 45 24 29 ab 31 29 3a 99:<8a>
>02d71 41 24 3b 3a 89 31 30 34:<2b>
>02d79 00 8f 2d 6a 00 8b d4 28:<be>
>02d81 45 53 24 2c 41 24 29 b2:<c7>
>02d89 30 a7 31 30 34 00 af 2d:<87>
>02d91 6b 00 8b c3 28 53 45 24:<43>
>02d99 29 b3 37 36 a7 53 45 24:<75>
>02da1 b2 53 45 24 aa 41 24 3a:<67>
>02da9 99 41 24 3b 3a 00 b8 2d:<2e>
>02db1 6c 00 89 31 30 34 00 ff:<b1>
>02db9 2d 6d 00 55 b2 30 3a 41:<a1>
>02dc1 24 b2 53 45 24 3a 81 49:<36>
>02dc9 b2 31 a4 c3 28 41 24 29:<a9>
>02dd1 3a 8b d4 28 41 24 2c 22:<67>
>02dd9 20 22 29 b3 b1 30 a7 41:<aa>
>02de1 24 b2 c9 28 41 24 2c c3:<c3>
>02de9 28 41 24 29 ab d4 28 41:<5a>
>02df1 24 2c 22 20 22 29 29 3a:<c0>
>02df9 55 b2 55 aa 31 00 39 2e:<bb>
>02e01 6e 00 82 3a 8b 55 b2 c3:<b7>
>02e09 28 53 45 24 29 a7 99 22:<1b>
>02e11 1d 22 3a fe 1a 30 2c 32:<56>
>02e19 2c 37 39 2c 32 32 3a 99:<65>
>02e21 22 1d 22 3a 9e 36 32 31:<68>
>02e29 32 3a 99 22 13 13 22 3a:<d3>
>02e31 5a b2 31 3a 89 38 31 00:<b0>
>02e39 6b 2e 6f 00 81 49 b2 31:<57>
>02e41 a4 c3 28 53 45 24 29 3a:<43>
>02e49 97 34 31 32 37 aa 49 2c:<33>
>02e51 c6 28 ca 28 53 45 24 2c:<4c>
>02e59 49 2c 31 29 29 3a 82 3a:<c7>
>02e61 97 34 31 32 37 aa 49 2c:<c5>
>02e69 30 00 bb 2e 70 00 99 22:<0a>
>02e71 1d 22 3a fe 1a 30 2c 32:<11>
>02e79 2c 37 39 2c 32 32 3a 99:<a5>
>02e81 22 1d 22 3a 9e 36 32 31:<a5>
>02e89 32 3a 9e 33 36 33 37 3a:<ac>
>02e91 97 32 36 30 33 2c 36 34:<f5>
>02e99 3a 99 43 24 22 53 22 3a:<ec>
>02ea1 9e 34 38 36 34 3a fe 09:<90>
>02ea9 41 3a 97 32 36 30 33 2c:<6d>
>02eb1 39 36 3a 99 43 24 22 55:<90>
>02eb9 22 00 ee 2e 71 00 9e 33:<04>
>02ec1 36 33 37 3a 8b 41 b2 32:<93>
>02ec9 a7 99 22 13 13 22 3a 89:<03>
>02ed1 37 38 3a d5 99 22 1d 22:<2e>
>02ed9 3a 9e 36 32 31 32 3a 99:<d3>
>02ee1 22 13 13 22 3a 5a b2 31:<2b>
>02ee9 3a 89 38 31 00 02 2f 72:<b9>
>02ef1 00 8f 20 2a 2a 2a 20 42:<01>
>02ef9 4c 4f 43 4b 20 2a 2a 2a:<11>
>02f01 00 0a 2f 73 00 5a b2 31:<0c>
>02f09 00 39 2f 74 00 50 31 b2:<43>
>02f11 33 31 3a 50 32 b2 34 3a:<d1>
>02f19 50 33 b2 34 39 3a 50 34:<48>
>02f21 b2 31 35 3a 55 24 b2 22:<43>
>02f29 c2 20 4c 20 4f 20 43 20:<8f>
>02f31 4b 22 3a 8d 33 30 38 00:<9d>
>02f39 4f 2f 75 00 99 22 1d 22:<46>
>02f41 3a fe 1a 33 32 2c 37 2c:<72>
>02f49 34 38 2c 31 35 00 92 2f:<1f>
>02f51 76 00 41 b2 34 3a 41 24:<c7>
>02f59 28 31 29 b2 22 c3 55 54:<36>
>02f61 22 3a 41 24 28 32 29 b2:<96>
>02f69 22 c3 4f 50 59 22 3a 41:<71>
>02f71 24 28 33 29 b2 22 d0 41:<8f>
>02f79 53 54 45 22 3a 41 24 28:<07>
```

```
>02f81 34 29 b2 22 d2 45 54 55:<11>
>02f89 52 4e 22 3a 8d 33 31 32:<17>
>02f91 00 9f 2f 77 00 41 b2 34:<96>
>02f99 3a 8d 33 31 35 00 dc 2f:<ef>
>02fa1 78 00 8b 5a b2 34 a7 99:<83>
>02fa9 22 1d 22 3a fe 1a 30 2c:<76>
>02fb1 32 2c 37 39 2c 32 32 3a:<54>
>02fb9 99 22 1d 22 3a 9e 36 32:<d1>
>02fc1 31 32 3a 99 22 13 13 22:<50>
>02fc9 43 24 22 45 22 43 24 22:<7b>
>02fd1 46 22 3a 5a b2 32 3a 89:<44>
>02fd9 38 31 00 f6 2f 79 00 97:<90>
>02fe1 32 36 30 33 2c 36 34 3a:<fa>
>02fe9 99 43 24 22 53 22 3a 9e:<fa>
>02ff1 33 36 33 37 00 20 30 7a:<7b>
>02ff9 00 8b 5a b2 33 a7 9e 35:<58>
>03001 38 37 34 3a 97 32 36 30:<c8>
>03009 33 2c 39 36 3a 99 43 24:<1d>
>03011 22 55 22 3a 9e 33 36 33:<dc>
>03019 37 3a 89 31 31 36 00 47:<b2>
>03021 30 7b 00 9e 35 32 32 30:<3a>
>03029 2c 5a 3a 97 32 36 30 33:<86>
>03031 2c 39 36 3a 99 43 24 22:<ef>
>03039 55 22 3a 9e 33 36 33 37:<33>
>03041 3a 89 31 31 36 00 5d 30:<1e>
>03049 7c 00 8f 20 2a 2a 2a 20:<dc>
>03051 53 50 45 43 49 41 4c 20:<c3>
>03059 2a 2a 2a 00 65 30 7d 00:<85>
>03061 5a b2 31 00 98 30 7e 00:<55>
>03069 50 31 b2 31 30 3a 50 32:<7a>
>03071 b2 36 3a 50 33 b2 33 30:<d8>
>03079 3a 50 34 b2 31 35 3a 55:<2d>
>03081 24 b2 22 d3 20 50 20 45:<70>
>03089 20 43 20 49 20 41 20 4c:<0c>
>03091 22 3a 8d 33 30 38 00 d2:<86>
>03099 30 7f 00 41 b2 33 3a 41:<ca>
>030a1 24 28 31 29 b2 22 d2 45:<eb>
>030a9 53 54 41 52 54 22 3a 41:<cd>
>030b1 24 28 32 29 b2 22 d2 45:<8c>
>030b9 53 45 54 22 3a 41 24 28:<8c>
>030c1 33 29 b2 22 d2 45 54 55:<96>
>030c9 52 4e 22 3a 8d 33 31 32:<a1>
>030d1 00 df 30 80 00 41 b2 33:<64>
>030d9 3a 8d 33 31 35 00 f2 30:<77>
>030e1 81 00 91 5a 89 31 33 30:<95>
>030e9 2c 31 33 31 2c 31 33 32:<a2>
>030f1 00 06 31 82 00 fe 1a 30:<56>
>030f9 2c 30 2c 37 39 2c 32 34:<89>
>03101 2c 31 3a 8a 00 11 31 83:<ea>
>03109 00 9e 35 37 33 34 34 00:<d1>
>03111 49 31 84 00 99 22 1d 22:<36>
>03119 3a fe 1a 30 2c 32 2c 37:<f2>
>03121 39 2c 32 32 3a 99 22 1d:<bc>
>03129 22 3a 9e 36 32 31 32 3a:<31>
>03131 99 22 13 13 22 43 24 22:<fa>
>03139 45 22 43 24 22 46 22 3a:<eb>
>03141 5a b2 37 3a 89 38 31 00:<e5>
>03149 5e 31 85 00 8f 20 2a 2a:<26>
>03151 2a 20 44 45 4c 45 54 45:<5c>
>03159 20 2a 2a 2a 00 66 31 86:<24>
>03161 00 5a b2 31 00 97 31 87:<3b>
>03169 00 50 31 b2 33 31 3a 50:<f5>
>03171 32 b2 33 3a 50 33 b2 34:<28>
>03179 39 3a 50 34 b2 32 30 3a:<4a>
>03181 55 24 b2 22 c4 20 45 20:<4a>
>03189 4c 20 45 20 54 20 45 22:<88>
>03191 3a 8d 33 30 38 00 ad 31:<3d>
>03199 88 00 99 22 1d 22 3a fe:<07>
>031a1 1a 33 32 2c 36 2c 34 38:<8c>
>031a9 2c 31 39 00 27 32 89 00:<0e>
>031b1 41 b2 37 3a 41 24 28 31:<1c>
>031b9 29 b2 22 d0 41 47 45 20:<46>
>031c1 31 22 3a 41 24 28 32 29:<81>
>031c9 b2 22 d0 41 47 45 20 32:<32>
>031d1 22 3a 41 24 28 33 29 b2:<8d>
>031d9 22 d0 41 47 45 20 33 22:<dd>
>031e1 3a 41 24 28 34 29 b2 22:<24>
>031e9 d0 41 47 45 20 34 22 3a:<cb>
>031f1 41 24 28 35 29 b2 22 d0:<7f>
>031f9 41 47 45 20 35 22 3a 41:<cc>
>03201 24 28 36 29 b2 22 d7 48:<13>
>03209 4f 4c 45 20 d4 45 58 54:<23>
>03211 22 3a 41 24 28 37 29 b2:<8d>
>03219 22 d2 45 54 55 52 4e 22:<e3>
>03221 3a 8d 33 31 32 00 34 32:<e6>
>03229 8a 00 41 b2 37 3a 8d 33:<18>
>03231 31 35 00 71 32 8b 00 8b:<ed>
>03239 5a b2 37 a7 99 22 1d 22:<8e>
>03241 3a fe 1a 30 2c 32 2c 37:<64>
>03249 39 2c 32 32 3a 99 22 1d:<6e>
>03251 22 3a 9e 36 32 31 32 3a:<60>
>03259 99 22 13 13 22 43 24 22:<2b>
>03261 45 22 43 24 22 46 22 3a:<9c>
>03269 5a b2 33 3a 89 38 31 00:<45>
>03271 ce 32 8c 00 8b 5a b2 36:<de>
>03279 a7 8d 33 32 35 3a 97 32:<9b>
>03281 35 30 2c 39 36 3a 97 32:<e6>
>03289 35 31 2c 32 38 3a 97 32:<a0>
>03291 35 32 2c 39 36 3a 97 32:<ca>
>03299 35 33 2c 32 38 3a 97 33:<43>
>032a1 39 36 38 2c 32 3a 97 33:<be>
>032a9 39 36 39 2c 30 3a 41 50:<13>
>032b1 b2 31 3a 8d 33 32 39 3a:<a3>
>032b9 e0 2c 32 30 2c 30 2c 22:<c8>
>032c1 30 31 22 3a 5a b2 37 3a:<89>
>032c9 89 31 33 39 00 4b 33 8d:<a0>
>032d1 00 8b 5a b2 41 50 a7 fe:<c8>
>032d9 18 3a 9e 33 36 33 37 3a:<b6>
>032e1 9e 36 38 39 31 2c 30 2c:<d5>
>032e9 33 32 2c 2c 2c 37 32 36:<fb>
>032f1 34 2c 31 31 38 32 34 3a:<c3>
>032f9 9e 36 38 39 31 2c 30 2c:<4c>
>03301 30 2c 2c 2c 31 31 38 32:<0f>
>03309 34 2c 31 36 33 38 34 3a:<a8>
>03311 9e 33 36 33 37 3a 4d b2:<72>
>03319 31 3a 97 33 39 36 38 2c:<3f>
>03321 32 3a 97 33 39 36 39 2c:<31>
>03329 30 3a 97 32 35 30 2c 39:<12>
>03331 36 3a 97 32 35 31 2c 32:<bd>
>03339 38 3a 97 32 35 32 2c 39:<7b>
>03341 36 3a 97 32 35 33 2c 32:<0d>
>03349 38 00 57 33 8e 00 8d 33:<9a>
>03351 32 39 3a fe 19 00 93 33:<22>
>03359 8f 00 9e 33 36 33 37 3a:<b9>
>03361 9e 36 38 39 31 2c 31 2c:<15>
>03369 33 32 2c 2c 2c 28 5a ab:<e2>
```

```
>03371 31 29 ac 39 31 32 30 aa:<c3>
>03379 31 30 32 34 2c 28 5a ab:<14>
>03381 31 29 ac 39 31 32 30 aa:<5c>
>03389 31 30 32 34 aa 34 35 31:<57>
>03391 39 00 d3 33 90 00 9e 36:<ef>
>03399 38 39 31 2c 31 2c 30 2c:<a1>
>033a1 2c 2c 28 5a ab 31 29 ac:<57>
>033a9 39 31 32 30 aa 31 30 32:<62>
>033b1 34 aa 34 35 32 30 2c 28:<d8>
>033b9 5a ab 31 29 ac 39 31 32:<7b>
>033c1 30 aa 31 30 32 34 aa 39:<49>
>033c9 31 31 39 3a 9e 33 36 33:<80>
>033d1 37 00 e0 33 91 00 5a b2:<67>
>033d9 37 3a 89 31 33 39 00 fb:<11>
>033e1 33 92 00 8f 20 2a 2a 2a:<27>
>033e9 2a 20 50 52 49 4e 54 20:<c7>
>033f1 54 45 58 54 20 2a 2a 2a:<db>
>033f9 2a 00 03 34 93 00 5a b2:<de>
>03401 31 00 32 34 94 00 50 31:<c7>
>03409 b2 33 31 3a 50 32 b2 36:<c6>
>03411 3a 50 33 b2 35 34 3a 50:<db>
>03419 34 b2 31 37 3a 55 24 b2:<64>
>03421 22 d0 20 52 20 49 20 4e:<14>
>03429 20 54 22 3a 8d 33 30 38:<0c>
>03431 00 48 34 95 00 99 22 1d:<ce>
>03439 22 3a fe 1a 33 32 2c 39:<1e>
>03441 2c 35 33 2c 31 36 00 a8:<28>
>03449 34 96 00 41 b2 34 3a 41:<f8>
>03451 24 28 31 29 b2 22 c6 49:<de>
>03459 52 53 54 20 d0 41 47 45:<32>
>03461 20 3a 22 aa c4 28 46 50:<e3>
>03469 29 3a 41 24 28 32 29 b2:<21>
>03471 22 cc 41 53 54 20 d0 41:<3d>
>03479 47 45 20 20 3a 22 aa c4:<e9>
>03481 28 4c 50 29 3a 41 24 28:<5b>
>03489 33 29 b2 22 c7 4f 20 4f:<9a>
>03491 4e 22 3a 41 24 28 34 29:<b5>
>03499 b2 22 d2 45 54 55 52 4e:<ec>
>034a1 22 3a 8d 33 31 32 00 b5:<91>
>034a9 34 97 00 41 b2 34 3a 8d:<d3>
>034b1 33 31 35 00 f2 34 98 00:<a9>
>034b9 8b 5a b2 34 a7 99 22 1d:<f4>
>034c1 22 3a fe 1a 30 2c 32 2c:<30>
>034c9 37 39 2c 32 32 3a 99 22:<73>
>034d1 1d 22 3a 9e 36 32 31 32:<f9>
>034d9 3a 99 22 13 13 22 43 24:<84>
>034e1 22 45 22 43 24 22 46 22:<df>
>034e9 3a 5a b2 35 3a 89 38 31:<d5>
>034f1 00 09 35 99 00 8b 5a b2:<eb>
>034f9 32 a7 31 35 39 3a d5 8b:<ea>
>03501 5a b2 33 a7 31 36 34 00:<9d>
>03509 35 35 9a 00 99 22 1d 22:<65>
>03511 3a fe 1a 33 32 2c 39 2c:<7c>
>03519 35 35 2c 31 36 3a e0 2c:<e9>
>03521 31 35 2c 31 2c 22 90 22:<08>
>03529 aa c4 28 46 50 29 3a a1:<05>
>03531 f9 41 24 00 6e 35 9b 00:<f3>
>03539 8b 41 24 b2 22 1d 22 af:<b7>
>03541 46 50 b3 35 a7 46 50 b2:<2a>
>03549 46 50 aa 31 3a 8b 46 50:<6d>
>03551 b1 4c 50 a7 4c 50 b2 4c:<ff>
>03559 50 aa 31 3a e0 2c 31 35:<09>
>03561 2c 33 2c 22 9f 22 aa c4:<c2>
>03569 28 4c 50 29 00 87 35 9c:<de>
>03571 00 8b 41 24 b2 22 9d 22:<f7>
>03579 af 46 50 b1 31 a7 46 50:<f1>
>03581 b2 46 50 ab 31 00 bb 35:<69>
>03589 9d 00 8b 41 24 b2 c7 28:<53>
>03591 31 33 29 a7 e0 2c 31 35:<17>
>03599 2c 31 2c 22 9f 22 aa c4:<73>
>035a1 28 46 50 29 3a fe 1a 33:<ba>
>035a9 32 2c 39 2c 35 33 2c 31:<b2>
>035b1 36 3a fe 19 3a 89 31 35:<9f>
>035b9 31 00 c4 35 9e 00 89 31:<77>
>035c1 35 34 00 f0 35 9f 00 99:<67>
>035c9 22 1d 22 3a fe 1a 33 32:<84>
>035d1 2c 39 2c 35 35 2c 31 36:<d1>
>035d9 3a e0 2c 31 35 2c 33 2c:<98>
>035e1 22 90 22 aa c4 28 4c 50:<0d>
>035e9 29 3a a1 f9 41 24 00 09:<10>
>035f1 36 a0 00 8b 41 24 b2 22:<f9>
>035f9 1d 22 af 4c 50 b3 35 a7:<f2>
>03601 4c 50 b2 4c 50 aa 31 00:<ae>
>03609 42 36 a1 00 8b 41 24 b2:<3b>
>03611 22 9d 22 af 4c 50 b1 31:<25>
>03619 a7 4c 50 b2 4c 50 ab 31:<4c>
>03621 3a 8b 4c 50 b3 46 50 a7:<b6>
>03629 46 50 b2 46 50 ab 31 3a:<2b>
>03631 e0 2c 31 35 2c 31 2c 22:<40>
>03639 9f 22 aa c4 28 46 50 29:<04>
>03641 00 76 36 a2 00 8b 41 24:<3f>
>03649 b2 c7 28 31 33 29 a7 e0:<24>
>03651 2c 31 35 2c 33 2c 22 9f:<2c>
>03659 22 aa c4 28 4c 50 29 3a:<64>
>03661 fe 1a 33 32 2c 39 2c 35:<70>
>03669 33 2c 31 36 3a fe 19 3a:<c8>
>03671 89 31 35 31 00 7f 36 a3:<88>
>03679 00 89 31 35 39 00 8b 36:<37>
>03681 a4 00 46 b2 5a 3a 5a b2:<cc>
>03689 31 00 dd 36 a5 00 50 31:<96>
>03691 b2 34 33 3a 50 32 b2 38:<e4>
>03699 3a 50 33 b2 36 37 3a 50:<02>
>036a1 34 b2 32 31 3a 55 24 b2:<73>
>036a9 22 d0 20 52 20 49 20 4e:<98>
>036b1 20 54 22 3a 8d 33 30 38:<d3>
>036b9 3a 4a 4f b2 b6 28 50 54:<e9>
>036c1 b2 31 29 ac 38 30 aa 38:<e6>
>036c9 30 ab 4c 43 ab 31 3a 8b:<26>
>036d1 4c 4d b1 4a 4f a7 4c 4d:<d6>
>036d9 b2 4a 4f 00 f4 36 a6 00:<dd>
>036e1 99 22 1d 22 3a fe 1a 34:<e0>
>036e9 34 2c 31 31 2c 36 36 2c:<0c>
>036f1 32 30 00 7b 37 a7 00 41:<5e>
>036f9 b2 35 3a 41 24 28 31 29:<0e>
>03701 b2 22 cc 45 46 54 20 4d:<1e>
>03709 41 52 47 49 4e 3a 20 22:<d1>
>03711 aa c9 28 22 20 22 aa c4:<d9>
>03719 28 4c 4d 29 2c 33 29 3a:<b6>
>03721 41 24 28 32 29 b2 22 d0:<6e>
>03729 52 49 4e 54 20 53 54 59:<2f>
>03731 4c 45 3a 20 22 aa c9 28:<ef>
>03739 22 20 22 aa c4 28 50 53:<e8>
>03741 29 2c 33 29 3a 41 24 28:<25>
>03749 33 29 b2 22 ce cc d1 3a:<ce>
>03751 22 aa c4 28 4e 4c 29 3a:<31>
>03759 41 24 28 34 29 b2 22 d0:<ee>
```

DAS SONDERANGEBOT: PRIVATE KLEINANZEIGEN KOSTENLOS!

Das bietet Ihnen 128 SPECIAL: KLEINANZEIGEN SIND KOSTENLOS FÜR PRIVATANBIETER! Suchen Sie etwas, haben Sie etwas zu verkaufen, zu tauschen, wollen Sie einen Club gründen? Coupon ausfüllen, auf Postkarte kleben oder in Briefumschlag stecken und abschicken. So einfach geht das. Wollen Sie das Heft nicht zerschneiden, können Sie den Coupon auch fotokopieren. Oder einfach den Anzeigentext uns so schicken, auf Postkarte oder im Brief. Aber bitte mit Druckbuchstaben oder in Schreibmaschinenschrift!

Und: Einschließlich Ihrer Adresse und/oder Telefonnummer sollten acht Zeilen à 28 Anschläge nicht überschritten werden.

ACHTUNG: WICHTIGER HINWEIS!

Wir veröffentlichen nur Kleinanzeigen privater Inserenten kostenlos. Gewerbliche Anzeigen kosten pro Zeile DM 4,80 plus Mehrwertsteuer!

Wir versenden für Privat-Inserenten keine Beleg-Exemplare!

Chiffre-Anzeigen sind nicht gestattet! Wir behalten uns vor, Anzeigen, die gegen rechtliche, sittliche oder sonstige Gebote verstoßen, abzulehnen!

Anzeigenabdruck in der Reihenfolge ihres Eingangs, kein Rechtsanspruch auf den Abdruck in der nächsten Ausgabe!

Die Insertion ist nicht vom Kauf des Heftes abhängig! Wir behalten uns vor, Anzeigen, die nicht zum Themenkreis des Heftes – Computer – gehören, nicht abzudrucken oder sie nur insoweit zu berücksichtigen, wie es der Umfang des kostenlosen Anzeigenteils zuläßt.

PROGRAMMSERVICE
Achtung! Preis-Senkung!!!

Hiermit bestelle ich in Kenntnis Ihrer Verkaufsbedingungen die Listings dieses Heftes auf

☐ Diskette (30 DM)

Name ____________________

Vorname ____________________

Straße/Hausnr. ____________________

PLZ/Ort ____________________

Ich bezahle:

☐ per beiliegendem Verrechnungsscheck / Bargeld

☐ bargeldlos per Bankeinzug von meinem Konto (nur möglich in der Bundesrepublik!)

bei (Bank) und Ort ____________________

Kontonummer ____________________

Bankleitzahl ____________________

(steht auf jedem Kontoauszug)

Verkaufsbedingungen: Lieferung nur gegen Vorkasse oder Bankabbuchung. Keine Nachnahme. Umtausch bei Nichtfunktionieren.

Unterschrift ____________________ 128/VI

Bitte ausschneiden und einsenden an

COMMODORE WELT
KASSETTENSERVICE 128/VI
POSTFACH 1161
D-8044 UNTERSCHLEISSHEIM

Jede Diskette nur noch 30 DM! Sie sparen 10 DM

```
>03761 52 49 4e 54 22 3a 41 24:<e1>
>03769 28 35 29 b2 22 d2 45 54:<7d>
>03771 55 52 4e 22 3a 8d 33 31:<f5>
>03779 32 00 88 37 a8 00 41 b2:<e6>
>03781 35 3a 8d 33 31 35 00 c6:<27>
>03789 37 a9 00 8b 5a b2 35 a7:<a3>
>03791 99 22 1d 22 3a fe 1a 30:<bd>
>03799 2c 32 2c 37 39 2c 32 32:<a3>
>037a1 3a 99 22 1d 22 3a 9e 36:<e2>
>037a9 32 31 32 3a 99 22 13 13:<cd>
>037b1 22 43 24 22 45 22 43 24:<3e>
>037b9 22 46 22 3a 5a b2 46 3a:<2c>
>037c1 89 31 34 38 00 d2 37 aa:<0d>
>037c9 00 8b 5a b2 31 a7 fe 18:<46>
>037d1 00 03 38 ab 00 99 22 1d:<41>
>037d9 22 3a fe 1a 34 34 2c 31:<2f>
>037e1 31 2c 36 36 2c 32 30 3a:<c0>
>037e9 e0 2c 31 36 2c 31 2c 22:<b5>
>037f1 90 22 aa c9 28 22 20 22:<bf>
>037f9 aa c4 28 4c 4d 29 2c 33:<fd>
>03801 29 00 0c 38 ac 00 a1 f9:<b4>
>03809 41 24 00 3b 38 ad 00 8b:<0b>
>03811 41 24 b2 22 1d 22 af 4c:<61>
>03819 4d b3 b6 28 50 54 b2 31:<89>
>03821 29 ac 38 30 aa 38 30 ab:<50>
>03829 4c 43 ab 31 a7 4c 4d b2:<25>
>03831 4c 4d aa 31 3a 89 31 37:<29>
>03839 31 00 59 38 ae 00 8b 41:<5d>
>03841 24 b2 22 9d 22 af 4c 4d:<61>
>03849 b1 30 a7 4c 4d b2 4c 4d:<75>
>03851 ab 31 3a 89 31 37 31 00:<53>
>03859 86 38 af 00 8b 41 24 b2:<5e>
>03861 c7 28 31 33 29 a7 e0 2c:<75>
>03869 31 36 2c 31 2c 22 9f 22:<9d>
>03871 aa c9 28 22 20 22 aa c4:<1a>
>03879 28 4c 4d 29 2c 33 29 3a:<f9>
>03881 89 31 36 38 00 92 38 b0:<f9>
>03889 00 89 31 37 32 3a fe 19:<3f>
>03891 00 9e 38 b1 00 8b 5a b2:<9f>
>03899 32 a7 fe 18 00 cf 38 b2:<64>
>038a1 00 99 22 1d 22 3a fe 1a:<85>
>038a9 34 34 2c 31 31 2c 36 36:<a3>
>038b1 2c 32 30 3a e0 2c 31 36:<61>
>038b9 2c 33 2c 22 90 22 aa c9:<42>
>038c1 28 22 20 22 aa c4 28 50:<fe>
>038c9 53 29 2c 33 29 00 d8 38:<4f>
>038d1 b3 00 a1 f9 41 24 00 f8:<61>
>038d9 38 b4 00 8b 41 24 b2 22:<ef>
>038e1 1d 22 af 50 53 b3 32 35:<e8>
>038e9 35 a7 50 53 b2 50 53 aa:<ba>
>038f1 31 3a 89 31 37 38 00 16:<e1>
>038f9 39 b5 00 8b 41 24 b2 22:<2f>
>03901 9d 22 af 50 53 b1 30 a7:<00>
>03909 50 53 b2 50 53 ab 31 3a:<77>
>03911 89 31 37 38 00 43 39 b6:<36>
>03919 00 8b 41 24 b2 c7 28 31:<bc>
>03921 33 29 a7 e0 2c 31 36 2c:<86>
>03929 33 2c 22 9f 22 aa c9 28:<3a>
>03931 22 20 22 aa c4 28 50 53:<d6>
>03939 29 2c 33 29 3a 89 31 36:<55>
>03941 38 00 4f 39 b7 00 89 31:<5d>
>03949 37 39 3a fe 19 00 6c 39:<55>
>03951 b8 00 8b 5a b2 33 a7 fe:<b6>
>03959 18 3a 8b 4e 4c b2 30 a7:<2b>
>03961 4e 4c b2 31 3a d5 4e 4c:<9c>
>03969 b2 30 00 97 39 b9 00 99:<41>
>03971 22 1d 22 3a fe 1a 34 34:<db>
>03979 2c 31 31 2c 36 36 2c 32:<2c>
>03981 30 3a e0 2c 37 2c 35 2c:<fd>
>03989 c4 28 4e 4c 29 3a fe 19:<d0>
>03991 3a 89 31 36 38 00 cf 39:<e3>
>03999 ba 00 9e 33 36 33 37 3a:<5a>
>039a1 9e 36 39 37 33 2c 30 2c:<57>
>039a9 31 2c 2c 2c 37 32 36 34:<87>
>039b1 2c 31 36 33 38 33 2c 28:<84>
>039b9 41 50 ab 31 29 ac 39 31:<b3>
>039c1 32 30 aa 31 30 32 34 3a:<e2>
>039c9 9e 33 36 33 37 00 15 3a:<09>
>039d1 bb 00 81 46 b2 46 50 a4:<51>
>039d9 4c 50 3a 52 46 b2 31 3a:<c6>
>039e1 9f 34 2c 44 4e 2c 37 3a:<97>
>039e9 98 34 2c 22 22 3b 3a a0:<36>
>039f1 34 3a 9f 34 2c 44 4e 2c:<86>
>039f9 31 30 3a a0 34 3a 52 46:<e1>
>03a01 b2 30 3a 99 22 1d 22 3a:<67>
>03a09 fe 1a 30 2c 32 2c 4c 43:<bd>
>03a11 2c 32 32 00 5b 3a bc 00:<a1>
>03a19 9e 33 36 33 37 3a 9e 36:<de>
>03a21 39 37 33 2c 31 2c 30 2c:<68>
>03a29 2c 2c 28 46 ab 31 29 ac:<5d>
>03a31 39 31 32 30 aa 31 30 32:<23>
>03a39 34 2c 28 46 ab 31 29 ac:<08>
>03a41 39 31 32 30 aa 31 30 32:<88>
>03a49 34 aa 39 31 31 39 2c 37:<a1>
>03a51 32 36 34 3a 9e 33 36 33:<45>
>03a59 37 00 65 3a bd 00 9e 33:<13>
>03a61 36 36 31 00 a1 3a be 00:<34>
>03a69 97 34 38 36 34 2c 44 4e:<5e>
>03a71 3a 97 34 38 36 35 2c 53:<d4>
>03a79 41 3a 97 34 38 36 36 2c:<cb>
>03a81 50 53 3a 97 34 38 36 37:<3c>
>03a89 2c 4e 4c 3a 97 34 38 36:<29>
>03a91 38 2c 4c 4d 3a 97 34 38:<04>
>03a99 36 39 2c 50 54 ab 31 00:<1f>
>03aa1 ab 3a bf 00 9e 34 38 37:<21>
>03aa9 30 00 b5 3a c0 00 9e 33:<63>
>03ab1 36 36 31 00 c5 3a c1 00:<c8>
>03ab9 9f 34 2c 44 4e 3a 98 34:<f4>
>03ac1 3a a0 34 00 15 3b c2 00:<30>
>03ac9 8b 46 b3 4c 50 a7 fe 18:<67>
>03ad1 3a 50 31 b2 37 3a 50 32:<b0>
>03ad9 b2 31 30 3a 50 33 b2 33:<bc>
>03ae1 36 3a 50 34 b2 31 34 3a:<74>
>03ae9 55 24 b2 22 c9 4e 53 45:<0f>
>03af1 52 54 20 4e 45 58 54 20:<3a>
>03af9 50 41 50 45 52 20 46 4f:<c5>
>03b01 52 20 50 41 47 45 22 aa:<6d>
>03b09 c4 28 46 aa 31 29 3a 8d:<d9>
>03b11 33 30 38 00 41 3b c3 00:<fd>
>03b19 e0 2c 31 2c 30 2c 22 d0:<a0>
>03b21 52 45 53 53 20 d2 45 54:<75>
>03b29 55 52 4e 20 4f 52 20 3c:<12>
>03b31 20 54 4f 20 c1 42 4f 52:<e1>
>03b39 54 22 3a a1 f9 41 24 00:<f0>
>03b41 97 3b c4 00 8b 41 24 b3:<0d>
>03b49 b1 22 3c 22 a7 31 39 38:<c0>
```

```
>03b51 3a d5 9e 33 36 33 37 3a:<37>
>03b59 9e 36 39 37 33 2c 31 2c:<31>
>03b61 30 2c 2c 2c 28 41 50 ab:<98>
>03b69 31 29 ac 39 31 32 30 aa:<bb>
>03b71 31 30 32 34 2c 28 41 50:<57>
>03b79 ab 31 29 ac 39 31 32 30:<9a>
>03b81 aa 31 30 32 34 aa 39 31:<18>
>03b89 31 39 2c 37 32 36 34 3a:<60>
>03b91 9e 33 36 33 37 00 cf 3b:<89>
>03b99 c5 00 99 22 1d 22 3a fe:<2a>
>03ba1 1a 30 2c 32 2c 37 39 2c:<c8>
>03ba9 32 32 3a 99 22 1d 22 3a:<44>
>03bb1 9e 36 32 31 32 3a 99 22:<8c>
>03bb9 13 13 22 43 24 22 45 22:<29>
>03bc1 43 24 22 46 22 3a 5a b2:<d5>
>03bc9 35 3a 89 38 31 00 ff 3b:<c8>
>03bd1 c6 00 55 24 b2 22 20 20:<8e>
>03bd9 20 20 20 20 20 20 20 20:<ae>
>03be1 20 20 20 20 20 20 20 20:<7e>
>03be9 20 20 20 20 20 20 20 20:<0d>
>03bf1 20 20 22 3a 8d 33 30 38:<ef>
>03bf9 3a 82 3a fe 19 00 47 3c:<c3>
>03c01 c7 00 9e 33 36 33 37 3a:<bd>
>03c09 9e 36 39 37 33 2c 31 2c:<0e>
>03c11 30 2c 2c 2c 28 41 50 ab:<b8>
>03c19 31 29 ac 39 31 32 30 aa:<a4>
>03c21 31 30 32 34 2c 28 41 50:<36>
>03c29 ab 31 29 ac 39 31 32 30:<fc>
>03c31 aa 31 30 32 34 aa 39 31:<38>
>03c39 31 39 2c 37 32 36 34 3a:<c0>
>03c41 9e 33 36 33 37 00 7f 3c:<96>
>03c49 c8 00 99 22 1d 22 3a fe:<8d>
>03c51 1a 30 2c 32 2c 37 39 2c:<69>
>03c59 32 32 3a 99 22 1d 22 3a:<25>
>03c61 9e 36 32 31 32 3a 99 22:<ad>
>03c69 13 13 22 43 24 22 45 22:<ca>
>03c71 43 24 22 46 22 3a 5a b2:<73>
>03c79 35 3a 89 38 31 00 92 3c:<93>
>03c81 c9 00 8f 20 2a 2a 2a 20:<b5>
>03c89 44 49 53 4b 20 2a 2a 2a:<aa>
>03c91 00 9a 3c ca 00 5a b2 31:<22>
>03c99 00 c7 3c cb 00 50 31 b2:<73>
>03ca1 31 33 3a 50 32 b2 33 3a:<53>
>03ca9 50 33 b2 33 34 3a 50 34:<3e>
>03cb1 b2 31 38 3a 55 24 b2 22:<95>
>03cb9 c4 20 49 20 53 20 4b 22:<f7>
>03cc1 3a 8d 33 30 38 00 44 3d:<84>
>03cc9 cc 00 41 b2 36 3a 41 24:<3b>
>03cd1 28 31 29 b2 22 c4 49 52:<30>
>03cd9 45 43 54 4f 52 59 22 3a:<d8>
>03ce1 41 24 28 32 29 b2 22 c6:<ef>
>03ce9 4f 52 4d 41 54 20 44 49:<fc>
>03cf1 53 4b 22 3a 41 24 28 33:<5d>
>03cf9 29 b2 22 cc 4f 41 44 20:<9e>
>03d01 54 45 58 54 22 3a 41 24:<ed>
>03d09 28 34 29 b2 22 d3 41 56:<c9>
>03d11 45 20 54 45 58 54 22 3a:<c1>
>03d19 41 24 28 35 29 b2 22 d3:<19>
>03d21 41 56 45 20 50 41 52 41:<53>
>03d29 4d 45 54 45 52 22 3a 41:<59>
>03d31 24 28 36 29 b2 22 d2 45:<e8>
>03d39 54 55 52 4e 22 3a 8d 33:<c6>
>03d41 31 32 00 51 3d cd 00 41:<05>
>03d49 b2 36 3a 8d 33 31 35 00:<c0>
>03d51 70 3d ce 00 91 5a 89 32:<81>
>03d59 30 37 2c 32 31 30 2c 32:<8d>
>03d61 34 32 2c 32 32 39 2c 32:<c4>
>03d69 31 38 2c 32 32 35 00 a7:<90>
>03d71 3d cf 00 50 31 b2 34 30:<ca>
>03d79 3a 50 32 b2 35 3a 50 33:<86>
>03d81 b2 37 30 3a 50 34 b2 32:<f2>
>03d89 30 3a 55 24 b2 22 c4 20:<cb>
>03d91 49 20 52 20 45 20 43 20:<2c>
>03d99 54 20 4f 20 52 20 59 22:<42>
>03da1 3a 8d 33 30 38 00 d6 3d:<bb>
>03da9 d0 00 99 43 24 22 4c 1d:<05>
>03db1 22 43 24 22 54 22 3b 3a:<1f>
>03db9 ee 3a 97 32 34 38 2c 31:<e8>
>03dc1 39 32 3a 8b 44 53 b3 b1:<4d>
>03dc9 30 a7 8d 32 32 36 3a d5:<b8>
>03dd1 a1 f9 41 24 00 01 3e d1:<b2>
>03dd9 00 99 22 1d 22 3a fe 1a:<67>
>03de1 30 2c 32 2c 37 39 2c 32:<69>
>03de9 32 3a 99 22 1d 22 3a 9e:<21>
>03df1 36 32 31 32 3a 99 22 13:<d9>
>03df9 13 22 3a 89 32 30 32 00:<4a>
>03e01 3d 3e d2 00 50 31 b2 31:<63>
>03e09 36 3a 50 32 b2 31 30 3a:<36>
>03e11 50 33 b2 34 30 3a 50 34:<b7>
>03e19 b2 32 30 3a 55 24 b2 22:<28>
>03e21 c6 20 4f 20 52 20 4d 20:<51>
>03e29 41 20 54 20 20 20 44 20:<83>
>03e31 49 20 53 20 4b 22 3a 8d:<c2>
>03e39 33 30 38 00 64 3e d3 00:<d3>
>03e41 41 b2 32 3a 41 24 28 31:<3c>
>03e49 29 b2 22 c3 41 4e 43 45:<02>
>03e51 4c 22 3a 41 24 28 32 29:<8a>
>03e59 b2 22 cf 4b 22 3a 8d 33:<5a>
>03e61 31 32 00 84 3e d4 00 e0:<80>
>03e69 2c 32 2c 36 2c 22 c9 4e:<53>
>03e71 53 45 52 54 20 44 49 53:<47>
>03e79 4b 2c 20 50 4c 45 41 53:<35>
>03e81 45 22 00 95 3e d5 00 5a:<1a>
>03e89 b2 31 3a 41 b2 32 3a 8d:<86>
>03e91 33 31 35 00 c9 3e d6 00:<93>
>03e99 8b 5a b2 31 a7 99 22 1d:<75>
>03ea1 22 3a fe 1a 30 2c 32 2c:<0b>
>03ea9 37 39 2c 32 32 3a 99 22:<1e>
>03eb1 1d 22 3a 9e 36 32 31 32:<9f>
>03eb9 3a 99 22 13 13 22 3a 5a:<6d>
>03ec1 b2 32 3a 89 32 30 33 00:<f7>
>03ec9 f1 3e d7 00 f1 22 53 43:<24>
>03ed1 52 49 50 54 43 41 4c 4c:<97>
>03ed9 20 20 44 41 54 41 22 2c:<f4>
>03ee1 49 42 43 3a 8b 44 53 b3:<8b>
>03ee9 b1 30 a7 8d 32 32 36 00:<fc>
>03ef1 20 3f d8 00 99 22 1d 22:<45>
>03ef9 3a fe 1a 30 2c 32 2c 37:<f0>
>03f01 39 2c 32 32 3a 99 22 1d:<f2>
>03f09 22 3a 9e 36 32 31 32 3a:<e2>
>03f11 99 22 13 13 22 3a 5a b2:<bc>
>03f19 32 3a 89 32 30 33 00 3a:<cd>
>03f21 3f d9 00 8f 20 2a 20 53:<e7>
>03f29 41 56 45 20 50 41 50 52:<bd>
>03f31 41 4d 45 54 45 52 20 2a:<cc>
>03f39 00 96 3f da 00 50 31 b2:<bd>
```

```
>03f41 32 35 3a 50 32 b2 39 3a:<84>
>03f49 50 33 b2 35 30 3a 50 34:<51>
>03f51 b2 31 34 3a 55 24 b2 22:<ce>
>03f59 c9 4e 53 45 52 54 20 4f:<99>
>03f61 52 49 47 49 4f 4e 41 4c:<58>
>03f69 20 44 49 53 4b 22 3a 8d:<bf>
>03f71 33 30 38 3a 99 22 11 20:<87>
>03f79 d3 41 56 45 20 50 41 52:<87>
>03f81 41 4d 45 54 45 52 20 28:<6c>
>03f89 59 2f 4e 29 20 3f 22 3a:<07>
>03f91 a1 f9 42 24 00 af 3f db:<32>
>03f99 00 8b 42 24 b3 b1 22 59:<c9>
>03fa1 22 af 42 24 b3 b1 22 5a:<23>
>03fa9 22 a7 32 32 34 00 c7 3f:<e8>
>03fb1 dc 00 fe 15 3a 8b 44 53:<be>
>03fb9 b3 b1 30 a7 8d 32 32 36:<6b>
>03fc1 3a 89 32 32 34 00 e3 3f:<01>
>03fc9 dd 00 f2 22 53 43 20 37:<b9>
>03fd1 22 3a 8b 44 53 b1 31 a7:<8e>
>03fd9 8d 32 32 36 3a 89 32 32:<38>
>03fe1 34 00 0a 40 de 00 9f 35:<80>
>03fe9 2c 38 2c 35 2c 22 53 43:<3c>
>03ff1 20 37 2c 50 2c 57 22 3a:<f2>
>03ff9 8b 44 53 b3 b1 30 a7 8d:<07>
>04001 32 32 36 3a 89 32 32 34:<0f>
>04009 00 4c 40 df 00 98 35 2c:<14>
>04011 4c 43 3a 98 35 2c 50 54:<e7>
>04019 3a 98 35 2c 44 4e 3a 98:<3f>
>04021 35 2c 53 41 3a 98 35 2c:<4d>
>04029 4c 4d 3a 98 35 2c 50 53:<5c>
>04031 3a 98 35 2c 4e 4c 3a fe:<92>
>04039 0f 3a 8b 44 53 b3 b1 30:<b4>
>04041 a7 8d 32 32 36 3a 89 32:<2b>
>04049 32 34 00 7b 40 e0 00 99:<39>
>04051 22 1d 22 3a fe 1a 30 2c:<c0>
>04059 32 2c 37 39 2c 32 32 3a:<c5>
>04061 99 22 1d 22 3a 9e 36 32:<91>
>04069 31 32 3a 99 22 13 13 22:<7e>
>04071 3a 5a b2 35 3a 89 32 30:<78>
>04079 33 00 b3 40 e1 00 99 22:<a3>
>04081 1d 22 3a fe 1a 30 2c 32:<a8>
>04089 2c 37 39 2c 32 32 3a 99:<d4>
>04091 22 1d 22 3a 9e 36 32 31:<e0>
>04099 32 3a 99 22 13 13 22 43:<1e>
>040a1 24 22 45 22 43 24 22 46:<45>
>040a9 22 3a 5a b2 34 3a 89 38:<35>
>040b1 31 00 ec 40 e2 00 50 31:<35>
>040b9 b2 32 30 3a 50 32 b2 39:<07>
>040c1 3a 50 33 b2 36 30 3a 50:<1a>
>040c9 34 b2 31 34 3a 55 24 b2:<d1>
>040d1 22 c4 20 49 20 53 20 4b:<c6>
>040d9 20 2d 20 45 20 52 20 52:<69>
>040e1 20 4f 20 52 22 3a 8d 33:<e1>
>040e9 30 38 00 10 41 e3 00 e0:<2f>
>040f1 2c 28 50 33 ab 50 31 ab:<6b>
>040f9 c3 28 44 53 24 29 29 ad:<83>
>04101 32 2c 31 2c 44 53 24 3a:<6d>
>04109 a1 f9 41 24 3a 8e 00 1f:<91>
>04111 41 e4 00 8f 20 2a 20 53:<80>
>04119 41 56 45 20 2a 00 4d 41:<6e>
>04121 e5 00 50 31 b2 33 30 3a:<a0>
>04129 50 32 b2 31 34 3a 50 33:<6b>
>04131 b2 34 35 3a 50 34 b2 31:<40>
>04139 39 3a 55 24 b2 22 d3 20:<8a>
>04141 41 20 56 20 45 22 3a 8d:<85>
>04149 33 30 38 00 69 41 e6 00:<b3>
>04151 99 22 1d 22 3a fe 1a 33:<56>
>04159 31 2c 31 38 2c 34 34 2c:<d9>
>04161 31 38 3a 8d 32 35 32 00:<77>
>04169 78 41 e7 00 8b 41 24 b2:<3a>
>04171 22 22 a7 32 34 30 00 b0:<08>
>04179 41 e8 00 9e 33 36 33 37:<9a>
>04181 3a 9e 36 39 37 33 2c 30:<fd>
>04189 2c 31 2c 2c 2c 37 32 36:<d6>
>04191 34 2c 31 36 33 38 33 2c:<6a>
>04199 28 41 50 ab 31 29 ac 39:<56>
>041a1 31 32 30 aa 31 30 32 34:<c1>
>041a9 3a 9e 33 36 33 37 00 e9:<1e>
>041b1 41 e9 00 9e 33 36 33 37:<0d>
>041b9 3a 9e 35 31 36 35 3a 41:<8d>
>041c1 b2 d1 28 c9 28 d2 28 c2:<a8>
>041c9 28 32 35 35 29 29 2c 32:<50>
>041d1 29 aa c9 28 d2 28 c2 28:<8c>
>041d9 32 35 34 29 29 2c 32 29:<68>
>041e1 29 3a 9e 33 36 33 37 00:<fa>
>041e9 01 42 ea 00 fe 15 3a 8b:<95>
>041f1 44 53 b3 b1 30 a7 8d 32:<c9>
>041f9 32 36 3a 89 32 34 30 00:<23>
>04201 28 42 eb 00 97 31 30 32:<49>
>04209 33 2c 4c 43 3a fe 10 28:<ba>
>04211 41 24 29 2c 42 31 2c 50:<26>
>04219 31 30 32 33 a4 50 28 41:<20>
>04221 29 3a 8d 33 32 33 00 8f:<aa>
>04229 42 ec 00 8b 44 53 b2 36:<29>
>04231 33 a7 fe 18 3a 50 31 b2:<80>
>04239 32 30 3a 50 32 b2 39 3a:<0e>
>04241 50 33 b2 35 34 3a 50 34:<bc>
>04249 b2 31 34 3a 55 24 b2 22:<aa>
>04251 d0 20 52 20 4f 20 42 20:<28>
>04259 4c 20 45 20 4d 22 3a 8d:<19>
>04261 33 30 38 3a 99 22 11 20:<e2>
>04269 c6 49 4c 45 20 45 58 49:<7f>
>04271 53 54 53 20 2d 20 4f 56:<92>
>04279 45 52 57 52 49 54 45 20:<ad>
>04281 28 59 2f 4e 29 20 3f 22:<d0>
>04289 3a a1 f9 42 24 00 cc 42:<22>
>04291 ed 00 8b 42 24 b2 22 59:<b8>
>04299 22 b0 42 24 b2 22 5a 22:<aa>
>042a1 a7 f2 28 41 24 29 3a 8b:<4c>
>042a9 44 53 b3 32 a7 32 33 35:<59>
>042b1 3a d5 8b 42 24 b2 22 59:<ce>
>042b9 22 b0 42 24 b2 22 5a 22:<02>
>042c1 a7 8d 32 32 36 3a 89 32:<e2>
>042c9 34 30 00 e8 42 ee 00 8b:<a4>
>042d1 42 24 b3 b1 22 5a 22 af:<62>
>042d9 42 24 b3 b1 22 59 22 a7:<0b>
>042e1 32 34 30 3a fe 19 00 f8:<a1>
>042e9 42 ef 00 8b 44 53 b3 b1:<f0>
>042f1 30 a7 8d 32 32 36 00 27:<7c>
>042f9 43 f0 00 99 22 1d 22 3a:<4a>
>04301 fe 1a 30 2c 32 2c 37 39:<71>
>04309 2c 32 32 3a 99 22 1d 22:<9d>
>04311 3a 9e 36 32 31 32 3a 99:<c0>
>04319 22 13 13 22 3a 5a b2 34:<be>
>04321 3a 89 32 30 33 00 36 43:<c0>
>04329 f1 00 8f 20 2a 20 4c 4f:<c6>
```

```
>04331 41 44 20 2a 00 64 43 f2:<22>
>04339 00 50 31 b2 33 30 3a 50:<df>
>04341 32 b2 31 34 3a 50 33 b2:<99>
>04349 34 35 3a 50 34 b2 31 39:<bf>
>04351 3a 55 24 b2 22 cc 20 4f:<fe>
>04359 20 41 20 44 22 3a 8d 33:<17>
>04361 30 38 00 80 43 f3 00 99:<e1>
>04369 22 1d 22 3a fe 1a 33 31:<89>
>04371 2c 31 38 2c 34 34 2c 31:<f7>
>04379 38 3a 8d 32 35 32 00 8f:<40>
>04381 43 f4 00 8b 41 24 b2 22:<3d>
>04389 22 a7 32 35 30 00 a7 43:<55>
>04391 f5 00 fe 15 3a 8b 44 53:<75>
>04399 b3 b1 30 a7 8d 32 32 36:<98>
>043a1 3a 89 32 35 30 00 cc 43:<19>
>043a9 f6 00 f5 28 41 24 29 a4:<bf>
>043b1 28 41 24 29 3a 8b 44 53:<b1>
>043b9 b2 36 33 a7 32 34 37 3a:<2d>
>043c1 d5 8d 32 32 36 3a 89 32:<2c>
>043c9 35 30 00 10 44 f7 00 8d:<e6>
>043d1 33 32 35 3a fe 15 3a 8b:<3e>
>043d9 44 53 b3 b1 30 a7 8d 32:<8e>
>043e1 32 36 3a 89 32 35 30 3a:<b8>
>043e9 d5 fe 11 28 41 24 29 2c:<03>
>043f1 42 31 2c 50 31 30 32 33:<c5>
>043f9 3a fe 02 31 35 3a 8b 44:<92>
>04401 53 b3 b1 30 a7 8d 32 32:<36>
>04409 36 3a 89 32 35 30 00 6b:<ca>
>04411 44 f8 00 4c 43 b2 c2 28:<5f>
>04419 31 30 32 33 29 3a 41 50:<0e>
>04421 b2 31 3a 99 22 13 13 22:<71>
>04429 3a e0 2c 32 30 2c 30 2c:<8a>
>04431 22 30 31 22 3a 97 32 35:<3c>
>04439 30 2c 39 36 3a 97 32 35:<6c>
>04441 31 2c 32 38 3a 97 32 35:<7b>
>04449 32 2c 39 36 3a 97 32 35:<86>
>04451 33 2c 32 38 3a 97 33 39:<bb>
>04459 36 38 2c 32 3a 97 33 39:<df>
>04461 36 39 2c 30 3a 8d 33 30:<93>
>04469 31 00 97 44 f9 00 9e 33:<e3>
>04471 36 33 37 3a 9e 36 39 37:<49>
>04479 33 2c 31 2c 30 2c 2c 2c:<49>
>04481 31 30 32 34 2c 31 30 31:<d3>
>04489 34 33 2c 37 32 36 34 3a:<a9>
>04491 9e 33 36 33 37 00 c5 44:<aa>
>04499 fa 00 99 22 1d 22 3a fe:<3e>
>044a1 1a 30 2c 32 2c 37 39 2c:<da>
>044a9 32 32 3a 99 22 1d 22 3a:<dd>
>044b1 9e 36 32 31 32 3a 8d 33:<db>
>044b9 32 39 3a 5a b2 33 3a 89:<57>
>044c1 32 30 33 00 db 44 fb 00:<39>
>044c9 8f 20 2a 2a 20 4c 53 2d:<13>
>044d1 49 4e 50 55 54 20 20 2a:<6e>
>044d9 2a 00 f9 44 fc 00 41 24:<a1>
>044e1 b2 22 22 3a 99 c7 28 31:<f6>
>044e9 34 32 29 3a 99 43 24 22:<2f>
>044f1 45 22 43 24 22 46 22 00:<9f>
>044f9 10 45 fd 00 a1 f9 5a 24:<4e>
>04501 3a 8b 5a 24 b2 c7 28 31:<62>
>04509 33 29 a7 32 35 38 00 42:<95>
>04511 45 fe 00 8b 5a 24 b2 c7:<f3>
>04519 28 32 30 29 a7 8b c3 28:<85>
>04521 41 24 29 b1 30 a7 41 24:<e3>
>04529 b2 c8 28 41 24 2c c3 28:<32>
>04531 41 24 29 ab 31 29 3a 99:<6d>
>04539 5a 24 3b 3a 89 32 35 33:<27>
>04541 00 57 45 ff 00 8b d4 28:<45>
>04549 4c 53 24 2c 5a 24 29 b2:<39>
>04551 30 a7 32 35 33 00 73 45:<40>
>04559 00 01 8b c3 28 41 24 29:<cd>
>04561 b3 31 33 a7 41 24 b2 41:<5c>
>04569 24 aa 5a 24 3a 99 5a 24:<1f>
>04571 3b 00 7c 45 01 01 89 32:<ae>
>04579 35 33 00 90 45 02 01 99:<77>
>04581 c7 28 31 34 29 3a 8b 41:<33>
>04589 24 b2 22 22 a7 8e 00 b8:<9a>
>04591 45 03 01 41 24 b2 c8 28:<86>
>04599 41 24 aa 22 20 20 20 20:<1a>
>045a1 20 20 20 20 20 20 20 20:<c3>
>045a9 20 22 2c 31 33 29 aa 22:<14>
>045b1 2e 53 43 22 3a 8e 00 cc:<aa>
>045b9 45 04 01 8f 20 2a 2a 2a:<6c>
>045c1 20 53 45 54 55 50 20 2a:<fa>
>045c9 2a 2a 00 d4 45 05 01 5a:<db>
>045d1 b2 31 00 03 46 06 01 50:<33>
>045d9 31 b2 32 30 3a 50 32 b2:<ad>
>045e1 35 3a 50 33 b2 34 31 3a:<42>
>045e9 50 34 b2 31 36 3a 55 24:<1b>
>045f1 b2 22 d3 20 45 20 54 20:<15>
>045f9 55 20 50 22 3a 8d 33 30:<8c>
>04601 38 00 5f 46 07 01 41 b2:<2f>
>04609 34 3a 41 24 28 31 29 b2:<e6>
>04611 22 d3 45 54 20 43 4f 4c:<53>
>04619 55 4d 4e 53 22 3a 41 24:<37>
>04621 28 32 29 b2 22 d0 52 49:<be>
>04629 4e 54 45 52 20 44 45 56:<7d>
>04631 49 43 45 22 3a 41 24 28:<e9>
>04639 33 29 b2 22 d0 52 49 4e:<f4>
>04641 54 45 52 20 54 59 50 45:<4c>
>04649 22 3a 41 24 28 34 29 b2:<dc>
>04651 22 d2 45 54 55 52 4e 22:<c7>
>04659 3a 8d 33 31 32 00 6c 46:<5e>
>04661 08 01 41 b2 34 3a 8d 33:<36>
>04669 31 35 00 a9 46 09 01 8b:<07>
>04671 5a b2 34 a7 99 22 1d 22:<52>
>04679 3a fe 1a 30 2c 32 2c 37:<a6>
>04681 39 2c 32 32 3a 99 22 1d:<5a>
>04689 22 3a 9e 36 32 31 32 3a:<a9>
>04691 5a b2 36 3a 99 22 13 13:<a1>
>04699 22 43 24 22 45 22 43 24:<a0>
>046a1 22 46 22 3a 89 38 31 00:<7f>
>046a9 bc 46 0a 01 91 5a 89 32:<02>
>046b1 36 37 2c 32 38 30 2c 32:<e0>
>046b9 39 34 00 fd 46 0b 01 4c:<8f>
>046c1 42 b2 4c 43 3a 50 31 b2:<17>
>046c9 32 33 3a 50 32 b2 38 3a:<f4>
>046d1 50 33 b2 35 38 3a 50 34:<f4>
>046d9 b2 32 30 3a 55 24 b2 22:<af>
>046e1 d3 20 45 20 54 20 20 20:<d1>
>046e9 43 20 4f 20 4c 20 55 20:<14>
>046f1 4d 20 4e 20 53 22 3a 8d:<c1>
>046f9 33 30 38 00 46 47 0c 01:<db>
>04701 41 b2 33 3a 41 24 28 31:<00>
>04709 29 b2 22 c3 4f 4c 55 4d:<11>
>04711 4e 53 3a 20 22 aa c9 28:<c0>
>04719 c4 28 4c 43 aa 31 29 2c:<0b>
```

```
>04721 32 29 3a 41 24 28 32 29:<09>
>04729 b2 22 c1 42 4f 52 54 22:<0e>
>04731 3a 41 24 28 33 29 b2 22:<12>
>04739 d2 45 54 55 52 4e 22 3a:<5f>
>04741 8d 33 31 32 00 95 47 0d:<24>
>04749 01 e0 2c 31 2c 37 2c 22:<f9>
>04751 d7 41 52 4e 49 4e 47 3a:<bc>
>04759 20 c9 46 20 59 4f 55 20:<a0>
>04761 20 2d d2 45 54 55 52 4e:<13>
>04769 2d 20 59 4f 55 27 4c 4c:<15>
>04771 20 20 20 20 20 20 20 20:<49>
>04779 20 20 20 4c 4f 4f 53 45:<57>
>04781 20 59 4f 55 52 20 43 55:<20>
>04789 52 52 45 4e 54 20 54 45:<71>
>04791 58 54 22 00 a6 47 0e 01:<4d>
>04799 41 b2 33 3a 5a b2 31 3a:<0d>
>047a1 8d 33 31 35 00 b9 47 0f:<70>
>047a9 01 91 5a 89 32 37 32 2c:<34>
>047b1 32 37 38 2c 32 37 39 00:<f5>
>047b9 e8 47 10 01 99 22 1d 22:<8a>
>047c1 3a fe 1a 32 34 2c 31 31:<f5>
>047c9 2c 35 37 2c 31 39 3a e0:<4c>
>047d1 2c 31 32 2c 31 2c 22 90:<ac>
>047d9 22 aa c9 28 c4 28 4c 42:<ff>
>047e1 aa 31 29 2c 32 29 00 f1:<b7>
>047e9 47 11 01 a1 f9 41 24 00:<9b>
>047f1 10 48 12 01 8b 41 24 b2:<17>
>047f9 22 1d 22 af 4c 42 b3 37:<61>
>04801 39 a7 4c 42 b2 4c 42 aa:<14>
>04809 31 3a 89 32 37 32 00 2f:<2e>
>04811 48 13 01 8b 4f 24 b2 22:<f2>
>04819 9d 22 af 4c 42 b1 31 39:<d0>
>04821 a7 4c 42 b2 4c 42 ab 31:<f6>
>04829 3a 89 32 37 32 00 5a 48:<0c>
>04831 14 01 8b 41 24 b2 c7 28:<ec>
>04839 31 33 29 a7 e0 2c 31 32:<c3>
>04841 2c 31 2c 22 9f 22 aa c9:<3b>
>04849 28 c4 28 4c 42 aa 31 29:<21>
>04851 2c 32 29 3a 89 32 37 30:<67>
>04859 00 63 48 15 01 89 32 37:<4d>
>04861 33 00 8e 48 16 01 99 22:<3b>
>04869 1d 22 3a fe 1a 30 2c 32:<68>
>04871 2c 37 39 2c 32 32 3a 99:<8c>
>04879 22 1d 22 3a 9e 36 32 31:<a6>
>04881 32 3a 99 22 13 13 22 3a:<99>
>04889 89 32 36 31 00 ef 48 17:<c5>
>04891 01 4c 43 b2 4c 42 3a 99:<23>
>04899 22 1d 22 3a fe 1a 30 2c:<1e>
>048a1 32 2c 37 39 2c 32 32 2c:<3b>
>048a9 31 3a 8d 33 32 35 3a 97:<61>
>048b1 32 35 30 2c 39 36 3a 97:<78>
>048b9 32 35 31 2c 32 38 3a 97:<bb>
>048c1 32 35 32 2c 39 36 3a 97:<9a>
>048c9 32 35 33 2c 32 38 3a 97:<86>
>048d1 33 39 36 38 2c 32 3a 97:<6d>
>048d9 33 39 36 39 2c 30 3a 8d:<c0>
>048e1 33 32 39 3a 8d 33 30 31:<de>
>048e9 3a 89 32 36 31 00 30 49:<0b>
>048f1 18 01 50 31 b2 32 33 3a:<b0>
>048f9 50 32 b2 38 3a 50 33 b2:<49>
>04901 35 33 3a 50 34 b2 31 37:<8e>
>04909 3a 55 24 b2 22 d0 20 52:<8e>
>04911 20 49 20 4e 20 54 20 45:<c1>
>04919 20 52 20 20 20 44 20 45:<aa>
>04921 20 56 20 49 20 43 20 45:<ea>
>04929 22 3a 8d 33 30 38 00 90:<3e>
>04931 49 19 01 41 b2 33 3a 41:<89>
>04939 24 28 31 29 b2 22 c4 45:<bc>
>04941 56 49 43 45 20 4e 55 4d:<21>
>04949 42 45 52 3a 22 aa c4 28:<71>
>04951 44 4e 29 3a 41 24 28 32:<da>
>04959 29 b2 22 d3 45 43 4f 4e:<d3>
>04961 44 41 52 59 20 41 44 44:<7d>
>04969 52 45 53 53 3a 20 22 aa:<53>
>04971 c9 28 c4 28 53 41 29 2c:<c8>
>04979 32 29 3a 41 24 28 33 29:<61>
>04981 b2 22 d2 45 54 55 52 4e:<db>
>04989 22 3a 8d 33 31 32 00 9c:<e0>
>04991 49 1a 01 41 b2 33 3a 5a:<10>
>04999 b2 31 00 a5 49 1b 01 8d:<44>
>049a1 33 31 35 00 b8 49 1c 01:<9d>
>049a9 91 5a 89 32 38 35 2c 32:<9a>
>049b1 38 37 2c 32 39 33 00 cd:<a9>
>049b9 49 1d 01 8b 44 4e b2 34:<6d>
>049c1 a7 44 4e b2 35 3a d5 44:<cb>
>049c9 4e b2 34 00 f6 49 1e 01:<73>
>049d1 99 22 1d 22 3a fe 1a 32:<b8>
>049d9 34 2c 31 31 2c 35 32 2c:<2d>
>049e1 31 36 3a e0 2c 31 37 2c:<c6>
>049e9 31 2c c4 28 44 4e 29 3a:<f7>
>049f1 89 32 38 32 00 23 4a 1f:<25>
>049f9 01 99 22 1d 22 3a fe 1a:<1c>
>04a01 32 34 2c 31 31 2c 35 32:<c0>
>04a09 2c 31 36 3a e0 2c 32 32:<f1>
>04a11 2c 33 2c 22 90 22 aa c9:<29>
>04a19 28 c4 28 53 41 29 2c 32:<2d>
>04a21 29 00 2c 4a 20 01 a1 f9:<fa>
>04a29 41 24 00 4b 4a 21 01 8b:<59>
>04a31 41 24 b2 22 1d 22 af 53:<88>
>04a39 41 b3 39 39 a7 53 41 b2:<c6>
>04a41 53 41 aa 31 3a 89 32 38:<83>
>04a49 37 00 69 4a 22 01 8b 41:<28>
>04a51 24 b2 22 9d 22 af 53 41:<b5>
>04a59 b1 30 a7 53 41 b2 53 41:<a4>
>04a61 ab 31 3a 89 32 38 37 00:<e7>
>04a69 96 4a 23 01 8b 41 24 b2:<ec>
>04a71 c7 28 31 33 29 a7 e0 2c:<e4>
>04a79 32 32 2c 33 2c 22 9f 22:<ac>
>04a81 aa c9 28 c4 28 53 41 29:<0f>
>04a89 2c 32 29 3a 5a b2 32 3a:<64>
>04a91 89 32 38 33 00 9f 4a 24:<ee>
>04a99 01 89 32 38 38 00 ce 4a:<c8>
>04aa1 25 01 99 22 1d 22 3a fe:<6a>
>04aa9 1a 30 2c 32 2c 37 39 2c:<e9>
>04ab1 32 32 3a 99 22 1d 22 3a:<a6>
>04ab9 9e 36 32 31 32 3a 99 22:<ff>
>04ac1 13 13 22 3a 5a b2 32 3a:<15>
>04ac9 89 32 36 32 00 0b 4b 26:<6c>
>04ad1 01 50 31 b2 32 33 3a 50:<30>
>04ad9 32 b2 38 3a 50 33 b2 35:<f4>
>04ae1 33 3a 50 34 b2 31 37 3a:<08>
>04ae9 55 24 b2 22 d0 20 52 20:<e7>
>04af1 49 20 4e 20 54 20 45 20:<b9>
>04af9 52 20 20 20 54 20 59 20:<06>
>04b01 50 20 45 22 3a 8d 33 30:<3c>
>04b09 38 00 53 4b 27 01 41 b2:<30>
```

```
>04b11 33 3a 41 24 28 31 29 b2:<30>
>04b19 22 c5 50 53 4f 4e 20 2d:<78>
>04b21 20 4b 4f 4d 50 41 54 49:<67>
>04b29 42 4c 45 22 3a 41 24 28:<b3>
>04b31 32 29 b2 22 cd d0 d3 20:<40>
>04b39 31 30 30 30 22 3a 41 24:<b5>
>04b41 28 33 29 b2 22 d2 45 54:<37>
>04b49 55 52 4e 22 3a 8d 33 31:<db>
>04b51 32 00 60 4b 28 01 41 b2:<93>
>04b59 33 3a 5a b2 50 54 00 69:<97>
>04b61 4b 29 01 8d 33 31 35 00:<c7>
>04b69 7c 4b 2a 01 8b 5a b2 33:<ee>
>04b71 a7 32 39 39 3a d5 50 54:<da>
>04b79 b2 5a 00 ab 4b 2b 01 99:<0f>
>04b81 22 1d 22 3a fe 1a 30 2c:<3d>
>04b89 32 2c 37 39 2c 32 32 3a:<af>
>04b91 99 22 1d 22 3a 9e 36 32:<35>
>04b99 31 32 3a 99 22 13 13 22:<f9>
>04ba1 3a 5a b2 33 3a 89 32 36:<02>
>04ba9 32 00 cc 4b 2c 01 8f 20:<9e>
>04bb1 2a 2a 2a 20 54 41 42 55:<d5>
>04bb9 4c 41 54 4f 52 45 4e 20:<4f>
>04bc1 53 45 54 5a 45 4e 20 2a:<75>
>04bc9 2a 2a 00 49 4c 2d 01 99:<93>
>04bd1 c7 28 31 34 32 29 3a e0:<e6>
>04bd9 2c 30 2c 31 2c c7 28 32:<c9>
>04be1 37 29 aa 22 51 9f 12 20:<30>
>04be9 92 22 aa c8 28 22 c2 c2:<af>
>04bf1 c2 c2 c2 c2 c2 c2 c2 c2:<5c>
>04bf9 c2 c2 c2 c2 c2 c2 c2 c2:<b2>
>04c01 c2 c2 c2 c2 c2 c2 c2 c2:<0f>
>04c09 c2 c2 c2 c2 c2 c2 c2 c2:<f4>
>04c11 c2 c2 c2 c2 c2 c2 c2 c2:<9b>
>04c19 c2 c2 c2 c2 c2 c2 c2 c2:<a2>
>04c21 c2 c2 c2 c2 c2 c2 c2 c2:<f9>
>04c29 c2 c2 c2 c2 c2 c2 c2 c2:<0b>
>04c31 c2 c2 c2 c2 c2 c2 c2 c2:<6c>
>04c39 c2 c2 c2 c2 22 2c 37 38:<68>
>04c41 29 aa 22 12 20 92 22 00:<a2>
>04c49 be 4c 2e 01 e0 2c 30 2c:<97>
>04c51 32 33 2c c7 28 32 37 29:<e6>
>04c59 aa 22 51 12 20 92 22 aa:<9d>
>04c61 c8 28 22 d9 d9 d9 d9 d9:<5e>
>04c69 d9 d9 d9 d9 d9 d9 d9 d9:<02>
>04c71 d9 d9 d9 d9 d9 d9 d9 d9:<0c>
>04c79 d9 d9 d9 d9 d9 d9 d9 d9:<ef>
>04c81 d9 d9 d9 d9 d9 d9 d9 d9:<47>
>04c89 d9 d9 d9 d9 d9 d9 d9 d9:<21>
>04c91 d9 d9 d9 d9 d9 d9 d9 d9:<e1>
>04c99 d9 d9 d9 d9 d9 d9 d9 d9:<b6>
>04ca1 d9 d9 d9 d9 d9 d9 d9 d9:<bf>
>04ca9 d9 d9 d9 d9 d9 d9 d9 d9:<75>
>04cb1 d9 22 2c 37 38 29 aa 22:<e8>
>04cb9 12 20 92 22 00 d8 4c 2f:<fb>
>04cc1 01 e0 2c 30 2c 31 3a eb:<62>
>04cc9 3a 8b 20 c2 28 32 33 36:<44>
>04cd1 29 b2 37 39 a7 ed 00 e8:<24>
>04cd9 4c 30 01 99 22 09 12 d8:<f7>
>04ce1 92 9d 22 3b 3a ec 00 03:<b6>
>04ce9 4d 31 01 e0 2c 30 2c 32:<6c>
>04cf1 33 3a eb 3a 8b 20 c2 28:<01>
>04cf9 32 33 36 29 b2 37 39 a7:<f8>
>04d01 ed 00 3c 4d 32 01 99 22:<8c>
>04d09 09 12 d6 92 9d 22 3b 3a:<d9>
>04d11 ec 3a 99 22 13 13 22 3a:<0e>
>04d19 e0 2c 4c 43 2c 31 2c 22:<37>
>04d21 12 20 92 22 3a e0 2c 4c:<5e>
>04d29 43 2c 32 33 2c 22 12 20:<0b>
>04d31 92 22 aa c7 28 31 34 29:<5a>
>04d39 3a 8e 00 5b 4d 33 01 8f:<11>
>04d41 20 20 2a 2a 2a 20 57 49:<a4>
>04d49 4e 44 4f 57 20 41 55 46:<32>
>04d51 42 41 55 45 4e 20 2a 2a:<38>
>04d59 2a 00 cd 4d 34 01 99 c7:<8f>
>04d61 28 31 34 32 29 3a fe 1a:<9a>
>04d69 50 31 2c 50 32 2c 50 33:<06>
>04d71 2c 50 34 2c 31 3a 99 22:<17>
>04d79 cf 22 c8 28 46 31 24 2c:<14>
>04d81 50 33 ab 50 31 ab 31 29:<ec>
>04d89 22 d0 9d 11 22 c8 28 46:<9d>
>04d91 34 24 2c 28 50 34 ab 50:<3e>
>04d99 32 ab 31 29 ac 33 29 3a:<17>
>04da1 99 22 13 11 22 c8 28 46:<d6>
>04da9 33 24 2c 28 50 34 ab 50:<76>
>04db1 32 ab 31 29 ac 33 29 22:<a6>
>04db9 cc 22 c8 28 46 32 24 2c:<a5>
>04dc1 50 33 ab 50 31 ab 31 29:<f7>
>04dc9 22 dc 22 00 07 4e 35 01:<21>
>04dd1 e0 2c 31 2c 32 2c c7 28:<4e>
>04dd9 31 34 29 aa c8 28 46 36:<c7>
>04de1 24 2c 50 33 ab 50 31 ab:<ed>
>04de9 31 29 3a e0 2c 28 50 33:<cd>
>04df1 ab 50 31 ab 31 ab c3 28:<ef>
>04df9 55 24 29 29 ad 32 aa 31:<1f>
>04e01 2c 31 2c 55 24 00 63 4e:<94>
>04e09 36 01 fe 1a 50 31 aa 31:<56>
>04e11 2c 50 34 aa 31 2c 50 33:<02>
>04e19 aa 31 2c 50 34 aa 31 3a:<0e>
>04e21 99 22 90 12 22 46 35 24:<04>
>04e29 3a fe 1a 50 33 aa 31 2c:<b7>
>04e31 50 32 aa 31 2c 50 33 aa:<29>
>04e39 31 2c 50 34 aa 31 3a 99:<06>
>04e41 22 12 22 46 35 24 22 9f:<a9>
>04e49 22 3a fe 1a 50 31 aa 31:<11>
>04e51 2c 50 32 aa 33 2c 50 33:<64>
>04e59 ab 31 2c 50 34 ab 31 3a:<32>
>04e61 8e 00 7d 4e 37 01 8f 20:<ae>
>04e69 2a 2a 2a 20 4d 45 4e 55:<8c>
>04e71 45 50 55 4e 4b 54 45 20:<9a>
>04e79 2a 2a 2a 00 e0 4e 38 01:<26>
>04e81 81 54 b2 30 a4 41 ab 31:<39>
>04e89 3a e0 2c 30 2c 31 aa 54:<f7>
>04e91 ac 32 2c 22 20 2d 20 22:<f1>
>04e99 aa c8 28 41 24 28 54 aa:<92>
>04ea1 31 29 aa 22 20 20 20 20:<54>
>04ea9 20 20 20 20 20 20 20 20:<15>
>04eb1 20 20 20 20 20 20 20 20:<d3>
>04eb9 20 20 20 20 20 20 20 20:<e7>
>04ec1 20 20 20 20 20 20 20 20:<58>
>04ec9 20 20 20 20 22 2c 50 33:<54>
>04ed1 ab 50 31 ab 37 29 aa 22:<09>
>04ed9 20 2d 20 22 3a 82 00 e6:<89>
>04ee1 4e 39 01 8e 00 fc 4e 3a:<ab>
>04ee9 01 8f 20 2a 2a 2a 20 41:<4d>
>04ef1 55 53 57 41 48 4c 20 2a:<86>
>04ef9 2a 2a 00 05 4f 3b 01 8d:<14>
```

```
>04f01 33 32 33 00 35 4f 3c 01:<39>
>04f09 e0 2c 31 2c 31 aa 28 5a:<d7>
>04f11 ab 31 29 ac 32 2c 22 90:<30>
>04f19 0f 2d 22 aa c7 28 32 37:<21>
>04f21 29 aa 22 4b 9d 2d 22 aa:<00>
>04f29 c7 28 32 37 29 aa c7 28:<3a>
>04f31 32 37 29 00 3e 4f 3d 01:<24>
>04f39 a1 f9 41 24 00 78 4f 3e:<0c>
>04f41 01 8b 41 24 b2 22 11 22:<0c>
>04f49 af 5a b3 41 a7 e0 2c 31:<04>
>04f51 2c 31 aa 28 5a ab 31 29:<da>
>04f59 ac 32 2c 22 9f 2d 22 aa:<2f>
>04f61 c7 28 32 37 29 aa 22 4b:<52>
>04f69 9d 2d 22 3a 5a b2 5a aa:<76>
>04f71 31 3a 89 33 31 36 00 b2:<c8>
>04f79 4f 3f 01 8b 41 24 b2 22:<cb>
>04f81 91 22 af 5a b1 31 a7 e0:<56>
>04f89 2c 31 2c 31 aa 28 5a ab:<2e>
>04f91 31 29 ac 32 2c 22 9f 2d:<f2>
>04f99 22 aa c7 28 32 37 29 aa:<68>
>04fa1 22 4b 9d 2d 22 3a 5a b2:<31>
>04fa9 5a ab 31 3a 89 33 31 36:<e1>
>04fb1 00 d6 4f 40 01 8b 41 24:<02>
>04fb9 b2 c7 28 31 33 29 a7 99:<b7>
>04fc1 22 13 13 9f 1d 22 3a fe:<2e>
>04fc9 1a 30 2c 32 2c 4c 43 2c:<1f>
>04fd1 32 32 3a 8e 00 df 4f 41:<0c>
>04fd9 01 89 33 31 37 00 f8 4f:<a9>
>04fe1 42 01 8f 20 2a 2a 2a 20:<97>
>04fe9 43 55 52 53 4f 52 20 4f:<f5>
>04ff1 55 54 20 2a 2a 2a 00 29:<bd>
>04ff9 50 43 01 97 32 31 2c 31:<ef>
>05001 3a 97 32 30 38 2c 32 3a:<4a>
>05009 97 38 34 32 2c 31 33 3a:<54>
>05011 97 38 34 33 2c 31 33 3a:<0d>
>05019 85 4a 24 3a a1 4a 24 3a:<ea>
>05021 97 32 31 2c 30 3a 8e 00:<9f>
>05029 45 50 44 01 8f 20 2a 2a:<b5>
>05031 2a 20 54 45 58 54 20 4c:<fa>
>05039 4f 45 53 43 48 45 4e 20:<cf>
>05041 2a 2a 2a 00 85 50 45 01:<35>
>05049 9e 33 36 33 37 3a 9e 36:<f1>
>05051 38 39 31 2c 30 2c 33 32:<15>
>05059 2c 2c 2c 37 32 36 34 2c:<6c>
>05061 31 31 38 32 34 3a 9e 36:<fb>
>05069 38 39 31 2c 30 2c 30 2c:<4e>
>05071 2c 2c 31 31 38 32 34 2c:<68>
>05079 31 36 33 38 34 3a 9e 33:<b0>
>05081 36 33 37 00 ba 50 46 01:<c1>
>05089 9e 33 36 33 37 3a 81 55:<84>
>05091 b2 31 30 32 34 a4 33 37:<d3>
>05099 35 30 34 a9 39 31 32 30:<0a>
>050a1 3a 9e 36 38 39 31 2c 31:<29>
>050a9 2c 33 32 2c 2c 2c 55 2c:<ae>
>050b1 55 aa 34 35 35 39 3a 82:<16>
>050b9 00 f0 50 47 01 81 55 b2:<d9>
>050c1 35 35 38 34 a4 34 32 30:<da>
>050c9 36 34 a9 39 31 32 30 3a:<7e>
>050d1 9e 36 38 39 31 2c 31 2c:<92>
>050d9 30 2c 2c 2c 55 2c 55 aa:<5f>
>050e1 34 35 35 39 3a 82 3a 9e:<ba>
>050e9 33 36 33 37 3a 8e 00 0f:<69>
>050f1 51 48 01 8f 20 2a 2a 2a:<e8>
>050f9 20 41 4e 5a 45 49 47 45:<b8>
>05101 20 41 55 46 20 4e 55 4c:<35>
>05109 4c 20 2a 2a 2a 00 29 51:<52>
>05111 49 01 99 22 13 13 1d 1d:<ed>
>05119 1d 1d 30 30 1d 1d 1d 1d:<84>
>05121 1d 1d 30 30 22 3a 8e 00:<a6>
>05129 3c 51 4a 01 8f 20 2a 2a:<59>
>05131 2a 20 54 52 41 50 20 2a:<1d>
>05139 2a 2a 00 8d 51 4b 01 8b:<fc>
>05141 45 52 b2 35 af 52 46 b2:<57>
>05149 31 a7 fe 18 3a 50 31 b2:<cc>
>05151 32 33 3a 50 32 b2 39 3a:<98>
>05159 50 33 b2 35 37 3a 50 34:<13>
>05161 b2 31 34 3a 55 24 b2 22:<77>
>05169 d0 20 52 20 49 20 4e 20:<2f>
>05171 54 20 2d 20 45 20 52 20:<77>
>05179 52 20 4f 20 52 22 3a 8d:<a3>
>05181 33 30 38 3a 52 46 b2 30:<fc>
>05189 3a a0 34 00 ba 51 4c 01:<13>
>05191 e0 2c 37 2c 31 2c 22 d0:<0e>
>05199 52 49 4e 54 45 52 20 4e:<a4>
>051a1 4f 54 20 50 52 45 53 45:<01>
>051a9 4e 54 22 3a a1 f9 41 24:<c5>
>051b1 3a d6 31 39 39 3a fe 19:<4f>
>051b9 00 05 52 4d 01 8b 45 52:<cf>
>051c1 b2 35 a7 fe 18 3a 97 32:<21>
>051c9 34 38 2c 31 39 32 3a 50:<a9>
>051d1 31 b2 32 30 3a 50 32 b2:<00>
>051d9 39 3a 50 33 b2 36 30 3a:<9e>
>051e1 50 34 b2 31 34 3a 55 24:<1a>
>051e9 b2 22 c4 20 49 20 53 20:<15>
>051f1 4b 20 2d 20 45 20 52 20:<a6>
>051f9 52 20 4f 20 52 22 3a 8d:<3b>
>05201 33 30 38 00 3b 52 4e 01:<d8>
>05209 e0 2c 36 2c 31 2c 22 30:<b6>
>05211 30 2c 20 44 45 56 49 43:<d0>
>05219 45 20 4e 4f 54 20 50 52:<5f>
>05221 45 53 45 4e 54 2c 30 30:<8f>
>05229 2c 30 30 22 3a a1 f9 41:<86>
>05231 24 3a d6 32 32 35 3a fe:<d9>
>05239 19 00 48 52 4f 01 8b 45:<bb>
>05241 52 b2 33 30 a7 d6 00 5b:<0f>
>05249 52 50 01 8f 20 2a 2a 2a:<ea>
>05251 20 48 45 4c 50 20 2a 2a:<66>
>05259 2a 00 be 52 51 01 50 31:<c2>
>05261 b2 30 3a 50 32 b2 32 3a:<08>
>05269 50 33 b2 37 38 3a 50 34:<64>
>05271 b2 32 31 3a 55 24 b2 22:<10>
>05279 c2 20 45 20 46 20 45 20:<18>
>05281 48 20 4c 20 53 20 bd 20:<f9>
>05289 42 20 45 20 52 20 53 20:<e6>
>05291 49 20 43 20 48 20 54 20:<cb>
>05299 20 20 2d 20 20 20 d3 20:<ae>
>052a1 43 20 52 20 49 20 50 20:<51>
>052a9 54 20 43 20 41 20 4c 20:<2d>
>052b1 4c 22 3a 8d 33 30 38 3a:<89>
>052b9 8d 33 32 33 00 17 53 52:<03>
>052c1 01 99 22 1d 11 22 43 24:<16>
>052c9 22 54 c8 cf cd c5 20 20:<fe>
>052d1 20 20 2d 20 c3 52 53 52:<27>
>052d9 20 49 4e 20 4c 49 4e 4b:<04>
>052e1 45 20 20 4f 42 45 52 45:<3d>
>052e9 20 c5 43 4b 45 20 20 20:<fb>
```

```
>052f1 20 c3 cc d2 20 20 20 20:<c0>
>052f9 20 2d 20 c3 52 53 52 20:<c6>
>05301 20 20 20 41 4e 20 20 20:<e0>
>05309 20 d3 45 49 54 45 4e 41:<b7>
>05311 4e 46 41 4e 47 00 69 53:<66>
>05319 53 01 99 22 c4 c5 cc 20:<8a>
>05321 20 20 20 20 2d 20 cc 49:<df>
>05329 4e 4b 45 53 20 20 da 45:<11>
>05331 49 43 48 45 4e 20 20 20:<82>
>05339 4c bc 53 43 48 45 4e 20:<18>
>05341 20 20 20 c9 ce d3 d4 20:<be>
>05349 20 20 20 2d 20 cc 45 45:<47>
>05351 52 5a 45 49 43 48 45 4e:<e8>
>05359 20 20 20 20 20 20 20 45:<c8>
>05361 49 4e 46 bd 47 45 4e 00:<86>
>05369 bb 53 54 01 99 22 20 20:<24>
>05371 20 20 20 20 20 20 2d 20:<98>
>05379 cf 42 45 52 45 20 20 20:<37>
>05381 da 45 49 4c 45 20 20 20:<86>
>05389 41 55 46 46 bd 4c 4c 45:<5f>
>05391 4e 20 20 20 20 d4 c1 c2:<07>
>05399 20 20 20 20 20 2d 20 d4:<97>
>053a1 41 42 55 4c 41 54 4f 52:<62>
>053a9 20 20 20 20 20 20 20 41:<c4>
>053b1 4e 53 50 52 49 4e 47 45:<c4>
>053b9 4e 00 0d 54 55 01 99 22:<da>
>053c1 d2 c5 d4 d5 d2 ce 20 20:<70>
>053c9 2d 20 c3 52 53 52 20 c1:<f6>
>053d1 4e 46 41 4e 47 20 4e bb:<09>
>053d9 43 48 53 54 45 20 da 45:<96>
>053e1 49 4c 45 20 20 20 20 d3:<cf>
>053e9 c8 c6 d4 2f d2 c5 d4 2d:<9d>
>053f1 20 c1 4b 54 55 45 4c 4c:<c4>
>053f9 45 20 20 20 da 45 49 4c:<32>
>05401 45 20 20 20 54 52 45 4e:<59>
>05409 4e 45 4e 00 5f 54 56 01:<90>
>05411 99 22 c6 31 20 2f 20 c6:<5a>
>05419 33 20 2d 20 d3 45 49 54:<e7>
>05421 45 20 48 4f 43 48 2f 52:<2b>
>05429 55 4e 54 45 52 53 43 52:<46>
>05431 4f 4c 4c 45 4e 20 20 20:<b1>
>05439 20 c6 35 20 2f 20 c6 37:<da>
>05441 20 2d 20 d7 4f 52 54 41:<fb>
>05449 4e 46 41 4e 47 20 20 20:<49>
>05451 20 20 20 41 4e 53 50 52:<c6>
>05459 49 4e 47 45 4e 00 b1 54:<6e>
>05461 57 01 99 22 c5 d3 c3 20:<62>
>05469 20 20 20 20 2d 20 cc 45:<88>
>05471 49 53 54 45 4e 4d 45 4e:<6a>
>05479 bd 20 20 20 20 20 20 41:<34>
>05481 4e 57 bb 48 4c 45 4e 20:<d9>
>05489 20 20 20 cc c9 ce c5 c6:<bd>
>05491 c5 c5 c4 2d 20 da 45 49:<eb>
>05499 4c 45 4e 56 4f 52 53 43:<2f>
>054a1 48 55 42 20 4f 48 4e 45:<28>
>054a9 20 d2 45 54 55 52 4e 00:<22>
>054b1 03 55 58 01 99 22 c3 d4:<21>
>054b9 d2 cc 2b 20 d1 20 2d 20:<7e>
>054c1 da 45 49 4c 45 20 20 41:<11>
>054c9 42 20 20 20 c3 52 53 52:<1f>
>054d1 20 20 4c bc 53 43 48 45:<2f>
>054d9 4e 20 20 20 20 c3 d4 d2:<66>
>054e1 cc 2b 20 d0 20 2d 20 da:<78>
>054e9 45 49 4c 45 20 20 42 49:<e1>
>054f1 53 20 20 20 c3 52 53 52:<9b>
>054f9 20 20 4c bc 53 43 48 45:<94>
>05501 4e 00 55 55 59 01 99 22:<e7>
>05509 20 20 20 20 20 20 c9 20:<f8>
>05511 2d 20 cc 45 45 52 5a 45:<5c>
>05519 49 4c 45 20 20 20 20 20:<1e>
>05521 20 20 20 45 49 4e 46 bd:<7e>
>05529 47 45 4e 20 20 20 20 20:<02>
>05531 20 20 20 20 20 c4 20 2d:<a2>
>05539 20 c1 4b 54 55 45 4c 4c:<85>
>05541 45 20 20 da 45 49 4c 45:<99>
>05549 20 20 45 4e 54 46 45 52:<85>
>05551 4e 45 4e 00 a7 55 5a 01:<d0>
>05559 99 22 20 20 20 20 20 20:<b1>
>05561 cc 20 2d 20 da 45 49 4c:<3f>
>05569 45 20 4c 49 4e 4b 53 20:<ec>
>05571 50 4f 53 49 54 49 4f 4e:<cd>
>05579 49 45 52 45 4e 20 20 20:<da>
>05581 20 20 20 20 20 20 20 d2:<a7>
>05589 20 2d 20 da 45 49 4c 45:<8f>
>05591 20 52 45 43 48 54 53 20:<83>
>05599 50 4f 53 49 54 49 4f 4e:<24>
>055a1 49 45 52 45 4e 00 f9 55:<b3>
>055a9 5b 01 99 22 20 20 20 20:<ab>
>055b1 20 20 cd 20 2d 20 da 45:<73>
>055b9 49 4c 45 20 20 20 20 20:<82>
>055c1 20 20 20 20 20 5a 45 4e:<76>
>055c9 54 52 49 45 52 45 4e 20:<e7>
>055d1 20 20 20 20 20 20 20 20:<24>
>055d9 d4 c1 c2 2d 20 d4 41 42:<26>
>055e1 55 4c 41 54 4f 52 20 53:<fd>
>055e9 45 54 5a 45 4e 20 2f 20:<37>
>055f1 4c bc 53 43 48 45 4e 00:<3d>
>055f9 4b 56 5c 01 99 22 20 20:<99>
>05601 20 20 20 20 ca 20 2d 20:<e6>
>05609 c3 52 53 52 20 20 20 41:<8b>
>05611 4e 20 20 20 20 da 45 49:<3a>
>05619 4c 45 4e 41 4e 46 41 4e:<83>
>05621 47 20 20 20 20 20 20 20:<9e>
>05629 20 20 20 cb 20 2d 20 c3:<d8>
>05631 52 53 52 20 20 20 20 20:<b1>
>05639 41 4e 20 20 20 20 20 da:<21>
>05641 45 49 4c 45 4e 45 4e 44:<c0>
>05649 45 00 9d 56 5d 01 99 22:<98>
>05651 20 20 20 20 20 20 da 20:<ce>
>05659 2d 20 c1 4c 4c 45 20 d4:<f9>
>05661 41 42 55 4c 41 54 4f 52:<46>
>05669 45 4e 20 20 4c bc 53 43:<32>
>05671 48 45 4e 20 20 20 20 20:<f3>
>05679 20 20 20 20 20 d9 20 2d:<c0>
>05681 20 d4 41 42 55 4c 41 54:<f5>
>05689 4f 52 45 4e 20 20 41 55:<66>
>05691 46 20 20 d3 54 41 4e 44:<f6>
>05699 41 52 44 00 ef 56 5e 01:<41>
>056a1 99 22 20 20 20 20 20 20:<5a>
>056a9 39 20 2d 20 d2 45 56 45:<be>
>056b1 52 53 4d 4f 44 55 53 20:<9b>
>056b9 20 20 20 20 45 49 4e 20:<c3>
>056c1 2f 20 41 55 53 20 20 20:<5d>
>056c9 20 20 20 20 20 20 20 d5:<bd>
>056d1 20 2d 20 d5 4e 54 45 52:<3c>
>056d9 53 54 52 45 49 43 48 45:<09>
```

```
>056e1 4e 20 20 20 20 20 45 49:<53>
>056e9 4e 2f 41 55 53 00 41 57:<c4>
>056f1 5f 01 99 22 20 20 20 20:<b7>
>056f9 20 20 b0 20 2d 20 d3 45:<7a>
>05701 49 54 45 20 20 41 42 20:<ae>
>05709 20 20 c3 52 53 52 20 20:<ef>
>05711 4c bc 53 43 48 45 4e 20:<3e>
>05719 20 20 20 20 20 20 20 20:<58>
>05721 20 5e 20 2d 20 d3 45 49:<50>
>05729 54 45 20 20 42 49 53 20:<08>
>05731 20 20 c3 52 53 52 20 20:<0a>
>05739 4c bc 53 43 48 45 4e 00:<8a>
>05741 93 57 60 01 99 22 20 20:<70>
>05749 20 20 20 20 2a 20 2d 20:<2a>
>05751 c1 4b 54 55 45 4c 4c 45:<a2>
>05759 20 20 d3 45 49 54 45 20:<74>
>05761 20 20 4c bc 53 43 48 45:<57>
>05769 4e 20 20 20 20 20 20 20:<4a>
>05771 20 20 20 c1 20 2d 20 c9:<2c>
>05779 4e 53 45 52 54 4d 4f 44:<d2>
>05781 55 53 20 20 20 20 20 20:<2d>
>05789 45 49 4e 20 2f 20 41 55:<7e>
>05791 53 00 9c 57 61 01 a1 f9:<a8>
>05799 41 24 00 ce 57 62 01 99:<7a>
>057a1 22 1d 22 3a fe 1a 30 2c:<e3>
>057a9 32 2c 37 39 2c 32 32 3a:<c6>
>057b1 99 22 1d 22 3a 9e 36 32:<54>
>057b9 31 32 3a 99 22 13 13 22:<7e>
>057c1 43 24 22 45 22 43 24 22:<8e>
>057c9 46 22 3a 8e 00 00 00 00:<f4>
```



SC 1 — 1 —

```
>00f78 2e 2c 3b 3a 2d 20 21 3f:<7b>
>00f80 00 00 00 00 00 00 00 00:<10>
>00f88 00 00 00 00 41 42 43 44:<6f>
>00f90 45 46 47 48 49 4a 4b 4c:<2f>
>00f98 4d 4e 4f 50 51 52 54 53:<dd>
>00fa0 55 56 57 58 59 5a c1 c2:<53>
>00fa8 c3 c4 c5 c6 c7 c8 c9 ca:<ba>
>00fb0 cb cc cd ce cf d0 d1 d2:<7a>
>00fb8 d3 d4 d5 d6 d7 d8 d9 da:<29>
>00fc0 31 32 33 34 35 36 37 38:<78>
>00fc8 39 30 21 40 24 25 26 2f:<88>
>00fd0 28 29 2b 2d 3d 3f 27 23:<4e>
>00fd8 5f ba b9 2a 5e ad 5b 5d:<98>
>00fe0 5c 3b 3a 2c 2e 3c 3e bc:<a5>
>00fe8 bb bd dc db dd be ff b0:<8a>
>00ff0 a5 a7 a3 a9 ab a6 aa a8:<a5>
>00ff8 a2 a4 a1 b1 22 20 00 00:<19>
```

SC 2 — 1 —

```
>01300 a9 f0 85 fe a9 24 85 ff:<8d>
>01308 a9 e8 85 fa a9 14 85 fb:<39>
>01310 20 8c 14 a9 00 85 fe a9:<84>
>01318 25 85 ff a9 f0 85 fa a9:<16>
>01320 14 85 fb 20 8c 14 a9 c0:<a2>
>01328 85 fe a9 25 85 ff a9 f8:<fe>
>01330 85 fa a9 14 85 fb 20 8c:<78>
>01338 14 a9 c0 85 fe a9 24 85:<d0>
>01340 ff a9 00 85 fa a9 15 85:<ce>
>01348 fb 20 8c 14 a9 10 85 fe:<5e>
>01350 a9 37 85 ff a9 08 85 fa:<92>
>01358 a9 15 85 fb 20 8c 14 a9:<41>
>01360 a0 85 fe a9 37 85 ff a9:<f2>
>01368 10 85 fa a9 15 85 fb 20:<da>
>01370 8c 14 a9 90 85 fe a9 37:<5c>
>01378 85 ff a9 18 85 fa a9 15:<71>
>01380 85 fb 20 8c 14 a9 d0 85:<91>
>01388 fe a9 35 85 ff a9 20 85:<3c>
>01390 fa a9 15 85 fb 20 8c 14:<98>
>01398 a9 c0 85 fe a9 2e 85 ff:<a3>
>013a0 a9 28 85 fa a9 15 85 fb:<3f>
>013a8 20 8c 14 a9 10 85 fe a9:<9e>
>013b0 3f 85 ff a9 30 85 fa a9:<86>
>013b8 15 85 fb 20 8c 14 a9 a0:<75>
>013c0 85 fe a9 3f 85 ff a9 38:<24>
>013c8 85 fa a9 15 85 fb 20 8c:<6e>
>013d0 14 a9 90 85 fe a9 3f 85:<e9>
>013d8 ff a9 40 85 fa a9 15 85:<80>
>013e0 fb 20 8c 14 a9 d0 85 fe:<c3>
>013e8 a9 3d 85 ff a9 48 85 fa:<c4>
>013f0 a9 15 85 fb 20 8c 14 a9:<92>
>013f8 20 85 fe a9 37 85 ff a9:<02>
>01400 50 85 fa a9 15 85 fb 20:<6d>
>01408 8c 14 a9 30 85 fe a9 37:<ca>
>01410 85 ff a9 58 85 fa a9 15:<58>
>01418 85 fb 20 8c 14 a9 40 85:<44>
```

```
>01420 fe a9 37 85 ff a9 60 85:<08>
>01428 fa a9 15 85 fb 20 8c 14:<c8>
>01430 a9 50 85 fe a9 37 85 ff:<d2>
>01438 a9 68 85 fa a9 15 85 fb:<f0>
>01440 20 8c 14 a9 60 85 fe a9:<6c>
>01448 37 85 ff a9 70 85 fa a9:<12>
>01450 15 85 fb 20 8c 14 a9 70:<5d>
>01458 85 fe a9 37 85 ff a9 78:<f4>
>01460 85 fa a9 15 85 fb 20 8c:<1e>
>01468 14 a9 80 85 fe a9 37 85:<7b>
>01470 ff a9 80 85 fa a9 15 85:<af>
>01478 fb 20 8c 14 a9 e0 85 fe:<03>
>01480 a9 36 85 ff a9 88 85 fa:<1e>
>01488 a9 15 85 fb a9 00 8d 00:<8f>
>01490 ff a0 00 20 e2 14 a9 12:<4a>
>01498 8d 00 d6 20 e2 14 a5 ff:<c4>
>014a0 8d 01 d6 20 e2 14 a9 13:<91>
>014a8 8d 00 d6 20 e2 14 a5 fe:<23>
>014b0 8d 01 d6 20 e2 14 a9 1f:<82>
>014b8 8d 00 d6 20 e2 14 b1 fa:<80>
>014c0 8d 01 d6 20 e2 14 a9 1e:<62>
>014c8 8d 00 d6 20 e2 14 a9 01:<d6>
>014d0 8d 01 d6 20 e2 14 e6 fe:<c6>
>014d8 d0 02 e6 ff c8 c0 08 d0:<8b>
>014e0 b5 60 2c 00 d6 10 fb 60:<c6>
>014e8 ff c0 c0 c0 c0 c0 c0 c0:<6d>
>014f0 ff 03 03 03 03 03 03 03:<f1>
>014f8 03 03 03 03 03 03 03 ff:<02>
>01500 c0 c0 c0 c0 c0 c0 c0 ff:<dd>
>01508 1c 22 1c 00 00 00 00 00:<f5>
>01510 0c 10 10 20 10 10 0c 00:<a8>
>01518 30 08 08 04 08 08 30 00:<53>
>01520 82 00 82 82 82 82 7c 00:<64>
>01528 ff 81 99 99 99 81 ff ff:<95>
>01530 e3 dd e3 ff ff ff ff ff:<a6>
>01538 f3 ef ef df ef ef f3 ff:<32>
>01540 cf f7 f7 fb f7 f7 cf ff:<87>
>01548 7d ff 7d 7d 7d 7d 83 ff:<a2>
>01550 00 00 00 00 3f ff fc 7c:<c6>
>01558 00 07 1c 70 c0 ff ff 0f:<ea>
>01560 c0 78 0e 03 00 0f ff f0:<6e>
>01568 00 00 00 80 7f ff fe 3c:<de>
>01570 3c 3c 3c 7c ff 7f 00 00:<36>
>01578 03 e3 03 0f fc e0 00 00:<0e>
>01580 c2 c3 c2 f0 1f 01 00 00:<fe>
>01588 3c fc 3e 3f ff fe 00 00:<f8>
>01590 fb 00 00 00 00 00 00 00:<31>
```


SC 3 — 1 —

```
>01300 a5 fa 8d 22 19 a5 fb 8d:<9c>
>01308 23 19 a5 fc 8d 24 19 a5:<43>
>01310 fd 8d 25 19 a9 60 85 fc:<5e>
>01318 a9 1c 85 fd a0 00 a2 00:<c3>
>01320 b1 fc 8c 28 19 dd 20 10:<b9>
>01328 d0 16 c8 c4 e7 f0 02 b0:<69>
>01330 0f e8 bd 20 10 c9 00 f0:<58>
>01338 47 b1 fc dd 20 10 f0 ea:<80>
>01340 ac 28 19 20 69 13 a5 fd:<c4>
>01348 c9 2e d0 04 a5 fc c9 30:<3c>
>01350 d0 cc ad 22 19 85 fa ad:<f6>
>01358 23 19 85 fb ad 24 19 85:<ea>
>01360 fc ad 25 19 85 fd a9 01:<57>
>01368 60 c8 c4 e7 90 11 f0 0f:<b8>
>01370 a0 00 18 a9 50 65 fc 85:<0c>
>01378 fc a9 00 65 fd 85 fd 60:<4b>
>01380 ac 28 19 a5 fc 8d 26 19:<21>
>01388 a5 fd 8d 27 19 38 ad 26:<09>
>01390 19 e9 60 8d 26 19 ad 27:<91>
>01398 19 e9 1c 8d 27 19 a2 00:<4a>
>013a0 ad 27 19 c9 00 d0 05 ad:<8b>
>013a8 26 19 c9 00 f0 15 38 ad:<a5>
>013b0 26 19 e9 50 8d 26 19 ad:<24>
>013b8 27 19 e9 00 8d 27 19 e8:<d9>
>013c0 4c a0 13 e0 0b b0 0f a9:<aa>
>013c8 60 85 fa a9 1c 85 fb 18:<60>
>013d0 20 18 c0 4c 0c 14 e0 2f:<31>
>013d8 90 17 a9 a0 85 fa a9 27:<7f>
>013e0 85 fb 8a 38 e9 2f 18 69:<ea>
>013e8 0b aa 18 20 18 c0 4c 0c:<2f>
>013f0 14 a5 fc 85 fa a5 fd 85:<24>
>013f8 fb 38 a5 fa e9 20 85 fa:<c6>
>01400 a5 fb e9 03 85 fb a2 0a:<8f>
>01408 18 20 18 c0 20 15 18 20:<95>
>01410 33 19 20 eb ee c9 20 d0:<75>
>01418 03 4c 40 13 c9 0d d0 f2:<9c>
>01420 a5 eb 8d 80 0f a5 ec 8d:<cc>
>01428 81 0f a9 02 60 a9 1f 85:<db>
>01430 fe a9 b6 85 ff a0 00 a2:<9d>
>01438 01 a9 fe 20 74 ff c9 20:<39>
>01440 f0 04 c9 00 d0 17 a5 ff:<68>
>01448 c9 04 d0 04 a5 fe c9 00:<11>
>01450 f0 0b a5 fe d0 02 c6 ff:<6b>
>01458 c6 fe 4c 35 14 e6 fe d0:<c2>
>01460 02 e6 ff 60 8d 28 19 a5:<18>
>01468 fa 8d 22 19 a5 fb 8d 23:<a9>
>01470 19 a5 fc 8d 24 19 a5 fd:<b2>
>01478 8d 25 19 a9 60 85 fa a9:<6e>
>01480 1c 85 fb a9 60 85 fc a9:<f7>
>01488 1c 85 fd 20 15 18 a9 13:<49>
>01490 20 d2 ff 20 33 19 20 eb:<cb>
>01498 ee c9 11 f0 03 4c d0 14:<40>
>014a0 a5 eb c9 16 d0 12 a5 fd:<cb>
>014a8 c9 2d d0 04 a5 fc c9 e0:<d4>
>014b0 f0 e4 20 41 1a 4c 93 14:<89>
>014b8 18 a9 50 65 fc 85 fc a9:<81>
```

```
>014c0 00 65 fd 85 fd a9 11 20:<d1>
>014c8 d2 ff 4c 93 14 4c 1c 15:<7f>
>014d0 c9 91 f0 03 4c 0a 15 a5:<05>
>014d8 eb c9 02 d0 15 a5 fd c9:<72>
>014e0 1c d0 04 a5 fc c9 60 d0:<5f>
>014e8 03 4c 96 14 20 dd 19 4c:<fd>
>014f0 93 14 38 a5 fc e9 50 85:<b2>
>014f8 fc a5 fd e9 00 85 fd a9:<2a>
>01500 91 20 d2 ff 4c 93 14 4c:<47>
>01508 1c 15 c9 0d f0 03 4c 1c:<37>
>01510 15 a5 fc 85 fe a5 fd 85:<26>
>01518 ff 4c 1f 15 4c 96 14 a9:<d2>
>01520 90 20 d2 ff 20 b1 1a 20:<13>
>01528 2e 19 20 eb ee c9 11 f0:<f1>
>01530 03 4c ab 15 a5 fc 8d 26:<24>
>01538 19 a5 fd 8d 27 19 38 ad:<bc>
>01540 26 19 e5 fe 8d 26 19 ad:<1f>
>01548 27 19 e5 ff 8d 27 19 a2:<bf>
>01550 01 ad 27 19 c9 00 d0 05:<7a>
>01558 ad 26 19 c9 00 f0 15 e8:<a0>
>01560 38 ad 26 19 e9 50 8d 26:<99>
>01568 19 ad 27 19 e9 00 8d 27:<00>
>01570 19 4c 51 15 e0 0c d0 03:<86>
>01578 4c 2a 15 a5 eb c9 16 d0:<d8>
>01580 12 a5 fd c9 2d d0 04 a5:<4c>
>01588 fc c9 e0 f0 9d 20 41 1a:<9f>
>01590 4c 1f 15 18 a9 50 65 fc:<81>
>01598 85 fc a9 00 65 fd 85 fd:<3f>
>015a0 a9 11 20 d2 ff 4c 1f 15:<fc>
>015a8 4c ef 16 c9 91 f0 03 4c:<9c>
>015b0 fc 15 a5 ff c5 fd d0 04:<0d>
>015b8 a5 fe c5 fc d0 03 4c 2a:<9f>
>015c0 15 a9 9f 20 d2 ff 20 b1:<5a>
>015c8 1a a5 eb c9 02 d0 15 a5:<36>
>015d0 fd c9 1c d0 04 a5 fc c9:<6c>
>015d8 60 d0 03 4c 2a 15 20 dd:<e0>
>015e0 19 4c 1f 15 38 a5 fc e9:<47>
>015e8 50 85 fc a5 fd e9 00 85:<84>
>015f0 fd a9 91 20 d2 ff 4c 27:<ab>
>015f8 15 4c ef 16 c9 0d f0 03:<cb>
>01600 4c ef 16 a0 00 18 a9 4f:<6e>
>01608 65 fc 85 fc a9 00 65 fd:<db>
>01610 85 fd a5 fc 8d 26 19 a5:<61>
>01618 fd 8d 27 19 a5 fe 8d 29:<c3>
>01620 19 a5 ff 8d 2a 19 a9 00:<0a>
>01628 85 fa a9 04 85 fb a9 00:<77>
>01630 85 fc a9 d8 85 fd b1 fe:<7b>
>01638 91 fa 18 a9 d0 65 fe 85:<a1>
>01640 fe a9 11 65 ff 85 ff b1:<86>
>01648 fe 91 fc 38 a5 fe e9 d0:<a5>
>01650 85 fe a5 ff e9 11 85 ff:<17>
>01658 ad 27 19 c5 ff d0 05 ad:<d6>
>01660 26 19 c5 fe f0 15 e6 fa:<48>
>01668 d0 02 e6 fb e6 fc d0 02:<fd>
>01670 e6 fd e6 fe d0 02 e6 ff:<a4>
>01678 4c 36 16 e6 fa d0 02 e6:<d3>
>01680 fb a9 00 91 fa ad 28 19:<bb>
>01688 c9 02 d0 03 4c d2 16 ad:<76>
>01690 29 19 85 fe ad 2a 19 85:<71>
>01698 ff a9 20 91 fe 18 a9 d0:<91>
>016a0 65 fe 85 fe a9 11 65 ff:<53>
>016a8 85 ff a9 00 91 fe 38 a5:<79>
>016b0 fe e9 d0 85 fe a5 ff e9:<5f>
>016b8 11 85 ff ad 27 19 c5 ff:<89>
>016c0 d0 05 ad 26 19 c5 fe f0:<f7>
>016c8 09 e6 fe d0 02 e6 ff 4c:<b5>
>016d0 99 16 ad 22 19 85 fa ad:<57>
>016d8 23 19 85 fb ad 24 19 85:<ee>
>016e0 fc ad 25 19 85 fd a9 9f:<fc>
>016e8 20 d2 ff 20 15 18 60 4c:<ba>
>016f0 2a 15 a5 fa 8d 22 19 a5:<64>
>016f8 fb 8d 23 19 a5 fc 8d 24:<af>
>01700 19 a5 fd 8d 25 19 a9 60:<e1>
>01708 85 fa a9 1c 85 fb a9 60:<35>
>01710 85 fc a9 1c 85 fd 20 15:<b2>
>01718 18 a9 13 20 d2 ff 20 33:<9c>
>01720 19 20 eb ee c9 11 f0 03:<f7>
>01728 4c 5b 17 a5 eb c9 16 d0:<20>
>01730 12 a5 fd c9 2d d0 04 a5:<2d>
>01738 fc c9 e0 f0 e4 20 41 1a:<b5>
>01740 4c 1e 17 18 a9 50 65 fc:<28>
>01748 85 fc a9 00 65 fd 85 fd:<60>
>01750 a9 11 20 d2 ff 4c 1e 17:<1e>
>01758 4c 12 18 c9 91 f0 03 4c:<c8>
>01760 95 17 a5 eb c9 02 d0 15:<d0>
>01768 a5 fd c9 1c d0 04 a5 fc:<2f>
>01770 c9 60 d0 03 4c 21 17 20:<15>
>01778 dd 19 4c 1e 17 38 a5 fc:<14>
>01780 e9 50 85 fc a5 fd e9 00:<cb>
>01788 85 fd a9 91 20 d2 ff 4c:<f8>
>01790 1e 17 4c 12 18 c9 0d f0:<1b>
>01798 03 4c 12 18 a0 00 a9 00:<5f>
>017a0 85 fa a9 04 85 fb a9 00:<75>
>017a8 85 fe a9 d8 85 ff b1 fa:<b0>
>017b0 c9 00 d0 03 4c fa 17 91:<aa>
>017b8 fc 18 a9 d0 65 fc 85 fc:<0f>
>017c0 a9 11 65 fd 85 fd b1 fe:<00>
>017c8 29 03 91 fc 38 a5 fc e9:<89>
>017d0 d0 85 fc a5 fd e9 11 85:<9a>
>017d8 fd e6 fa d0 02 e6 fb e6:<86>
>017e0 fc d0 02 e6 fd e6 fe d0:<c1>
>017e8 02 e6 ff a5 fd c9 2e d0:<34>
>017f0 04 a5 fc c9 30 f0 03 4c:<dd>
>017f8 ae 17 ad 22 19 85 fa ad:<1b>
>01800 23 19 85 fb ad 24 19 85:<b3>
>01808 fc ad 25 19 85 fd 20 15:<55>
>01810 18 60 4c 21 17 a5 eb 8d:<c8>
>01818 85 0f a5 ec 8d 86 0f a9:<a1>
>01820 13 20 d2 ff a9 1b 20 d2:<c9>
>01828 ff 20 d2 ff a2 00 8e 87:<a7>
>01830 0f a0 00 20 b2 18 b1 fa:<17>
>01838 cd 87 0f f0 06 8d 87 0f:<79>
>01840 20 6f 18 20 c0 18 b1 fa:<21>
>01848 20 d2 ff c4 e7 f0 04 c8:<d9>
>01850 4c 33 18 e0 14 f0 07 e8:<bf>
>01858 20 ea 18 4c 31 18 20 14:<97>
```

```
>01860 19 ad 85 0f 85 eb ad 86:<7d>
>01868 0f 85 ec 20 95 18 60 a9:<2c>
>01870 1b 20 d2 ff 20 d2 ff ad:<e5>
>01878 87 0f 29 01 c9 01 d0 06:<39>
>01880 a9 12 20 d2 ff 60 ad 87:<e0>
>01888 0f 29 02 c9 02 d0 05 a9:<0f>
>01890 02 20 d2 ff 60 8d 89 0f:<5c>
>01898 8e 8a 0f 8c 8b 0f a6 eb:<f4>
>018a0 a4 ec ca ca 18 20 18 c0:<97>
>018a8 ae 8a 0f ac 8b 0f ad 89:<be>
>018b0 0f 60 18 a9 d0 65 fa 85:<37>
>018b8 fa a9 11 65 fb 85 fb 60:<71>
>018c0 38 a5 fa e9 d0 85 fa a5:<51>
>018c8 fb e9 11 85 fb 60 18 a9:<3d>
>018d0 d0 65 fc 85 fc a9 11 65:<32>
>018d8 fd 85 fd 60 38 a5 fc e9:<4c>
>018e0 d0 85 fc a5 fd e9 11 85:<5d>
>018e8 fd 60 18 a9 50 65 fa 85:<bf>
>018f0 fa a9 00 65 fb 85 fb 60:<3f>
>018f8 18 a9 50 65 fc 85 fc a9:<fa>
>01900 00 65 fd 85 fd 60 38 a5:<96>
>01908 fc e9 50 85 fc a5 fd e9:<6c>
>01910 00 85 fd 60 38 a5 fa e9:<44>
>01918 40 85 fa a5 fb e9 06 85:<39>
>01920 fb 60 00 00 00 00 00 00:<13>
>01928 00 00 00 00 00 00 a9 9f:<e2>
>01930 20 d2 ff a5 eb 8d 85 0f:<2f>
>01938 a5 ec 8d 86 0f a9 1b 20:<61>
>01940 d2 ff 20 d2 ff a5 e7 8d:<7d>
>01948 8b 0f a9 13 20 d2 ff 20:<9a>
>01950 d2 ff ad 8b 0f 85 e7 a9:<48>
>01958 1d a2 04 20 d2 ff ca d0:<94>
>01960 fa a5 fc 8d 89 0f a5 fd:<95>
>01968 8d 8a 0f 38 ad 89 0f e9:<ca>
>01970 60 8d 89 0f ad 8a 0f e9:<53>
>01978 1c 8d 8a 0f a2 00 ad 8a:<ad>
>01980 0f c9 00 d0 05 ad 89 0f:<ea>
>01988 c9 00 f0 15 38 ad 89 0f:<1e>
>01990 e9 50 8d 89 0f ad 8a 0f:<11>
>01998 e9 00 8d 8a 0f e8 4c 7e:<79>
>019a0 19 8a 20 fb f9 a8 8a 20:<d5>
>019a8 d2 ff 98 20 d2 ff a9 1d:<53>
>019b0 a2 06 20 d2 ff ca d0 fa:<51>
>019b8 ad 86 0f 20 fb f9 a8 8a:<c0>
>019c0 20 d2 ff 98 20 d2 ff a9:<e9>
>019c8 02 85 e5 a9 16 85 e4 ad:<10>
>019d0 85 0f 85 eb ad 86 0f 85:<bd>
>019d8 ec 20 95 18 60 a5 fb c9:<cb>
>019e0 1c d0 04 a5 fa c9 60 d0:<84>
>019e8 01 60 38 a5 fa e9 50 85:<a3>
>019f0 fa a5 fb e9 00 85 fb 20:<70>
>019f8 06 19 a9 57 20 1e c0 a5:<2a>
>01a00 eb 8d 85 0f a5 ec 8d 86:<fb>
>01a08 0f a9 13 20 d2 ff a0 00:<c6>
>01a10 8c 87 0f 20 b2 18 b1 fa:<59>
>01a18 cd 87 0f f0 06 8d 87 0f:<3b>
>01a20 20 6f 18 20 c0 18 b1 fa:<e0>
>01a28 20 d2 ff c4 e7 f0 04 c8:<99>
>01a30 4c 13 1a ad 85 0f 85 eb:<8b>
>01a38 ad 86 0f 85 ec 20 95 18:<d7>
>01a40 60 a5 fb c9 27 d0 04 a5:<b6>
>01a48 fa c9 a0 d0 01 60 20 ea:<4c>
>01a50 18 20 f8 18 a9 56 20 1e:<1e>
>01a58 c0 a5 eb 8d 85 0f a5 ec:<6f>
>01a60 8d 86 0f a9 16 85 eb a9:<15>
>01a68 00 85 ec 20 95 18 18 a9:<0d>
>01a70 40 65 fa 85 fa a9 06 65:<80>
>01a78 fb 85 fb a0 00 8c 87 0f:<d1>
>01a80 20 b2 18 b1 fa cd 87 0f:<06>
>01a88 f0 06 8d 87 0f 20 6f 18:<2a>
>01a90 20 c0 18 b1 fa 20 d2 ff:<58>
>01a98 c4 e7 f0 04 c8 4c 80 1a:<e0>
>01aa0 ad 85 0f 85 eb ad 86 0f:<c7>
>01aa8 85 ec 20 95 18 20 14 19:<ec>
>01ab0 60 a5 eb 8d 2c 19 a5 ec:<33>
>01ab8 8d 2d 19 a9 4a 20 1e c0:<ab>
>01ac0 a0 00 20 ce 18 b1 fc 8d:<ac>
>01ac8 87 0f 20 6f 18 20 dc 18:<d1>
>01ad0 b1 fc 20 d2 ff c4 e7 f0:<8d>
>01ad8 04 c8 4c c2 1a ad 2c 19:<8d>
>01ae0 85 eb ad 2d 19 85 ec 20:<a0>
>01ae8 95 18 60 8d 22 19 8e 23:<cd>
>01af0 19 a5 fa 8d 26 19 a5 fb:<e0>
>01af8 8d 27 19 20 0f 88 84 fa:<6b>
>01b00 85 fb 20 0f 88 8c 24 19:<1e>
>01b08 8d 25 19 a0 00 a9 fa 8d:<be>
>01b10 b9 02 ae 22 19 ad 23 19:<38>
>01b18 20 77 ff ad 25 19 c5 fb:<1f>
>01b20 d0 05 ad 24 19 c5 fa f0:<ad>
>01b28 09 e6 fa d0 02 e6 fb 4c:<e7>
>01b30 0b 1b ad 26 19 85 fa ad:<97>
>01b38 27 19 85 fb 60 8d 22 19:<9f>
>01b40 8e 23 19 a5 fa 8d 24 19:<c6>
>01b48 a5 fb 8d 25 19 a5 fc 8d:<89>
>01b50 28 19 a5 fd 8d 29 19 20:<dc>
>01b58 0f 88 84 fa 85 fb 20 0f:<00>
>01b60 88 8c 26 19 8d 27 19 20:<e8>
>01b68 0f 88 84 fc 85 fd a0 00:<d7>
>01b70 a9 fa ae 22 19 20 74 ff:<52>
>01b78 a2 fc 8e b9 02 ae 23 19:<f0>
>01b80 20 77 ff ad 27 19 c5 fb:<c6>
>01b88 d0 05 ad 26 19 c5 fa f0:<aa>
>01b90 0f e6 fa d0 02 e6 fb e6:<0b>
>01b98 fc d0 02 e6 fd 4c 6e 1b:<a3>
>01ba0 ad 24 19 85 fa ad 25 19:<e3>
>01ba8 85 fb ad 28 19 85 fc ad:<76>
>01bb0 29 19 85 fd 60 86 77 77:<61>
```

SC 4

```
>00b00 20 14 0e 20 2d 1b 20 1f:<5a>
>00b08 0e 20 c4 18 c0 01 f0 06:<54>
>00b10 a9 07 20 d2 ff 60 a5 fd:<94>
>00b18 c9 2d d0 04 a5 fc c9 e0:<a9>
>00b20 f0 f3 a9 49 20 1e c0 20:<c3>
>00b28 2a 0e a9 90 85 fc a9 2d:<4d>
```

```
>00b30 85 fd a0 00 b1 fc aa 20:<71>
>00b38 f8 0d 8a 91 fc 18 a9 80:<f0>
>00b40 65 fc 85 fc a9 11 65 fd:<67>
>00b48 85 fd b1 fc aa 20 f8 0d:<f0>
>00b50 8a 91 fc 38 a5 fc e9 20:<24>
>00b58 85 fc a5 fd e9 12 85 fd:<27>
>00b60 c4 e7 f0 04 c8 4c 34 0b:<b7>
>00b68 ad 86 0f c5 fd d0 05 ad:<57>
>00b70 85 0f c5 fc f0 06 20 06:<4c>
>00b78 0e 4c 32 0b a0 ff c8 a9:<f3>
>00b80 20 91 fc 20 dc 0d a9 00:<09>
>00b88 91 fc 20 ea 0d c4 e7 d0:<a0>
>00b90 ed ad 89 0f 85 eb ad 8a:<23>
>00b98 0f 85 ec 20 2d 1b 60 20:<63>
>00ba0 14 0e 20 2d 1b 20 1f 0e:<da>
>00ba8 a5 fd c9 2d d0 04 a5 fc:<2c>
>00bb0 c9 e0 d0 03 4c 24 0c a9:<ed>
>00bb8 44 20 1e c0 20 2a 0e 20:<d1>
>00bc0 f8 0d a0 00 b1 fc aa 20:<22>
>00bc8 06 0e 8a 91 fc 18 a9 20:<6e>
>00bd0 65 fc 85 fc a9 12 65 fd:<63>
>00bd8 85 fd b1 fc aa 20 06 0e:<05>
>00be0 8a 91 fc 38 a5 fc e9 80:<ef>
>00be8 85 fc a5 fd e9 11 85 fd:<68>
>00bf0 c4 e7 f0 04 c8 4c c4 0b:<db>
>00bf8 a5 fd c9 2d d0 04 a5 fc:<cd>
>00c00 c9 e0 f0 03 4c bf 0b a9:<b2>
>00c08 e0 85 fc a9 2d 85 fd a0:<4f>
>00c10 ff c8 a9 20 91 fc 20 dc:<ee>
>00c18 0d a9 00 91 fc 20 ea 0d:<0b>
>00c20 c4 e7 d0 ed a5 fa 85 fc:<71>
>00c28 a5 fb 85 fd 18 a9 40 65:<ce>
>00c30 fc 85 fc a9 06 65 fd 85:<37>
>00c38 fd a9 16 85 eb 20 2d 1b:<50>
>00c40 20 3f 0d ad 85 0f 85 fc:<4c>
>00c48 ad 86 0f 85 fd 60 20 14:<ac>
>00c50 0e 20 2d 1b a9 50 20 1e:<f3>
>00c58 c0 a4 ec a9 20 91 fc 20:<fc>
>00c60 dc 0d a9 00 91 fc 20 ea:<eb>
>00c68 0d c0 00 f0 04 88 4c 5b:<5a>
>00c70 0c 60 20 14 0e 20 2d 1b:<56>
>00c78 a9 51 20 1e c0 a4 ec a9:<89>
>00c80 20 91 fc 20 dc 0d a9 00:<08>
>00c88 91 fc 20 ea 0d c4 e7 f0:<78>
>00c90 04 c8 4c 7f 0c 60 20 14:<a8>
>00c98 0e 20 2d 1b 20 c0 0c c0:<14>
>00ca0 01 f0 03 20 f0 0c 60 20:<e3>
>00ca8 14 0e 20 2d 1b a5 e7 38:<93>
>00cb0 e5 ec 8d 88 0f 20 c0 0c:<8f>
>00cb8 c0 01 f0 03 20 62 0d 60:<58>
>00cc0 a0 00 b1 fc c9 20 d0 0b:<da>
>00cc8 c4 e7 d0 03 4c b2 0d c8:<c9>
>00cd0 4c c2 0c 8c 86 0f a4 e7:<a0>
>00cd8 a2 00 b1 fc c9 20 d0 05:<4d>
>00ce0 88 e8 4c da 0c 8a 8c 85:<78>
>00ce8 0f 18 6d 86 0f a0 00 60:<97>
>00cf0 c5 ec 90 4a ac 86 0f a2:<07>
>00cf8 00 b1 fc 9d 20 10 20 dc:<2e>
>00d00 0d b1 fc 9d 72 10 20 ea:<6a>
>00d08 0d cc 85 0f f0 05 c8 e8:<d1>
>00d10 4c f9 0c e8 a9 00 9d 20:<ac>
>00d18 10 a0 00 20 7f 0c a4 ec:<d7>
>00d20 a2 00 bd 20 10 c9 00 f0:<11>
>00d28 12 91 fc 20 dc 0d bd 72:<c8>
>00d30 10 91 fc 20 ea 0d c8 e8:<b7>
>00d38 4c 22 0d 20 3f 0d 60 a9:<3c>
>00d40 4a 20 1e c0 a0 00 20 dc:<ca>
>00d48 0d b1 fc 8d 87 0f 20 9e:<12>
>00d50 18 20 ea 0d b1 fc 20 d2:<91>
>00d58 ff c4 e7 f0 04 c8 4c 46:<d0>
>00d60 0d 60 cd 88 0f 90 4a ac:<5e>
>00d68 85 0f a2 00 b1 fc 9d 20:<d9>
>00d70 10 20 dc 0d b1 fc 9d 72:<80>
>00d78 10 20 ea 0d cc 86 0f f0:<87>
>00d80 05 88 e8 4c 6c 0d e8 a9:<4b>
>00d88 00 9d 20 10 a0 00 20 7f:<22>
>00d90 0c a4 ec a2 00 bd 20 10:<e7>
>00d98 c9 00 f0 12 91 fc 20 dc:<e6>
>00da0 0d bd 72 10 91 fc 20 ea:<a6>
>00da8 0d 88 e8 4c 95 0d 20 3f:<d0>
>00db0 0d 60 a0 01 60 20 14 0e:<cc>
>00db8 20 2d 1b 20 c0 0c c0 01:<62>
>00dc0 f0 19 a8 a2 00 c0 00 f0:<6c>
>00dc8 12 e8 88 f0 06 88 f0 03:<f0>
>00dd0 4c c9 0d 86 ec 20 2d 1b:<50>
>00dd8 20 f4 0c 60 18 a9 d0 65:<57>
>00de0 fc 85 fc a9 11 65 fd 85:<48>
>00de8 fd 60 38 a5 fc e9 d0 85:<bb>
>00df0 fc a5 fd e9 11 85 fd 60:<00>
>00df8 18 a9 50 65 fc 85 fc a9:<78>
>00e00 00 65 fd 85 fd 60 38 a5:<b5>
>00e08 fc e9 50 85 fc a5 fd e9:<59>
>00e10 00 85 fd 60 ad 80 0f 85:<79>
>00e18 eb ad 81 0f 85 ec 60 a5:<3d>
>00e20 eb 8d 89 0f a5 ec 8d 8a:<ed>
>00e28 0f 60 a5 fc 8d 85 0f a5:<20>
>00e30 fd 8d 86 0f 60 20 d4 0e:<0d>
>00e38 a9 00 85 fe a9 10 85 ff:<46>
>00e40 a9 50 8d f1 0e a9 1c 8d:<55>
>00e48 f2 0e 4c 62 0e 20 d4 0e:<c9>
>00e50 a9 60 85 fe a9 19 85 ff:<7c>
>00e58 a9 9a 8d f1 0e a9 19 8d:<db>
>00e60 f2 0e a0 00 a9 00 8d 00:<5d>
>00e68 ff 20 ce 0e b1 fa 8d 84:<03>
>00e70 0f a2 12 a5 ff 20 cc cd:<fb>
>00e78 a2 13 a5 fe 20 cc cd a2:<e9>
>00e80 1f 20 da cd 91 fa 20 ce:<f3>
>00e88 0e a2 12 a5 ff 20 cc cd:<6d>
>00e90 a2 13 a5 fe 20 cc cd a2:<39>
>00e98 1f ad 84 0f 20 cc cd e6:<f3>
>00ea0 fe d0 02 e6 ff e6 fa d0:<37>
>00ea8 02 e6 fb a5 fb cd f2 0e:<2b>
>00eb0 d0 05 a5 fa cd f1 0e d0:<02>
>00eb8 b3 ad 88 0f 85 fa ad 89:<59>
>00ec0 0f 85 fb ad 8a 0f 85 fc:<ec>
>00ec8 ad 8b 0f 85 fd 60 2c 00:<aa>
```

```
>00ed0 d6 10 fb 60 a5 fa 8d 88:<85>
>00ed8 0f a5 fb 8d 89 0f a5 fc:<01>
>00ee0 8d 8a 0f a5 fd 8d 8b 0f:<9d>
>00ee8 a9 00 85 fa a9 13 85 fb:<26>
>00ef0 60 00 00 a2 00 00 00 00:<5c>
```

SC 5

```
>01300 04 07 00 00 00 00 ad 00:<90>
>01308 13 8d 56 16 ad 01 13 8d:<cf>
>01310 57 16 ad 02 13 8d 58 16:<1c>
>01318 ad 03 13 8d 59 16 ad 04:<74>
>01320 13 8d 5a 16 a9 04 ae 56:<3b>
>01328 16 ac 57 16 20 ba ff a9:<1c>
>01330 00 20 bd ff 20 c0 ff a2:<49>
>01338 04 20 c9 ff ad 05 13 c9:<fb>
>01340 01 d0 03 4c 7a 16 20 7d:<c3>
>01348 ff 1b 40 00 20 7d ff 1b:<b8>
>01350 21 00 ea ea ea ad 58 16:<e4>
>01358 20 d2 ff ad 59 16 c9 01:<34>
>01360 d0 09 20 7d ff 1b 78 01:<0c>
>01368 00 ea ea 20 2a 16 20 1c:<d5>
>01370 16 a5 fa 8d 5b 16 a5 fb:<b6>
>01378 8d 5c 16 a5 fc 8d 5d 16:<96>
>01380 a5 fd 8d 5e 16 a9 60 85:<25>
>01388 fa a9 1c 85 fb a9 30 85:<95>
>01390 fc a9 2e 85 fd ae 5a 16:<4d>
>01398 e0 00 f0 09 a9 20 20 d2:<af>
>013a0 ff ca 4c 98 13 a0 00 a9:<6e>
>013a8 00 8d 5f 16 ad 5f 16 d1:<50>
>013b0 fc f0 08 b1 fc 8d 5f 16:<f3>
>013b8 20 36 16 b1 fa a2 00 dd:<b1>
>013c0 60 16 f0 0b e8 e0 1a d0:<80>
>013c8 f6 20 d2 ff 4c be 15 c9:<40>
>013d0 b0 f0 03 4c de 13 a9 40:<65>
>013d8 20 d2 ff 4c be 15 c9 ad:<ef>
>013e0 f0 03 4c ed 13 a9 5c 20:<d0>
>013e8 d2 ff 4c be 15 c9 a5 f0:<77>
>013f0 03 4c fc 13 a9 ab 20 d2:<0a>
>013f8 ff 4c be 15 c9 a7 f0 03:<3a>
>01400 4c 0b 14 a9 b3 20 d2 ff:<2e>
>01408 4c be 15 c9 a3 f0 03 4c:<95>
>01410 1a 14 a9 b1 20 d2 ff 4c:<c0>
>01418 be 15 c9 a9 f0 03 4c 29:<9f>
>01420 14 a9 b2 20 d2 ff 4c be:<34>
>01428 15 c9 a8 f0 03 4c 38 14:<dc>
>01430 a9 b0 20 d2 ff 4c be 15:<ff>
>01438 c9 aa f0 03 4c 47 14 a9:<e4>
>01440 ae 20 d2 ff 4c be 15 c9:<68>
>01448 a2 f0 03 4c 56 14 a9 ad:<7d>
>01450 20 d2 ff 4c be 15 c9 a4:<91>
>01458 f0 03 4c 65 14 a9 bd 20:<cc>
>01460 d2 ff 4c be 15 c9 ab f0:<11>
>01468 03 4c 74 14 a9 db 20 d2:<d9>
>01470 ff 4c be 15 c9 a6 f0 03:<ee>
>01478 4c 83 14 a9 c0 20 d2 ff:<26>
>01480 4c be 15 c9 5f f0 03 4c:<52>
>01488 92 14 a9 a4 20 d2 ff 4c:<c5>
>01490 be 15 c9 bb f0 03 4c aa:<5d>
>01498 14 20 7d ff 1b 52 02 7b:<3d>
>014a0 1b 52 0b 00 ea ea ea 4c:<bb>
>014a8 be 15 c9 bc f0 03 4c c2:<a9>
>014b0 14 20 7d ff 1b 52 02 7c:<ed>
>014b8 1b 52 0b 00 ea ea ea 4c:<ed>
>014c0 be 15 c9 bd f0 03 4c da:<e4>
>014c8 14 20 7d ff 1b 52 02 7d:<12>
>014d0 1b 52 0b 00 ea ea ea 4c:<1e>
>014d8 be 15 c9 be f0 03 4c f2:<f2>
>014e0 14 20 7d ff 1b 52 02 7e:<43>
>014e8 1b 52 0b 00 ea ea ea 4c:<cd>
>014f0 be 15 c9 db f0 03 4c 0a:<13>
>014f8 15 20 7d ff 1b 52 02 5b:<6e>
>01500 1b 52 0b 00 ea ea ea 4c:<7b>
>01508 be 15 c9 dc f0 03 4c 22:<5a>
>01510 15 20 7d ff 1b 52 02 5c:<1d>
>01518 1b 52 0b 00 ea ea ea 4c:<ab>
>01520 be 15 c9 dd f0 03 4c 3a:<96>
>01528 15 20 7d ff 1b 52 02 5d:<45>
>01530 1b 52 0b 00 ea ea ea 4c:<5b>
>01538 be 15 c9 b1 f0 03 4c 52:<79>
>01540 15 20 7d ff 1b 52 06 5b:<f3>
>01548 1b 52 0b 00 ea ea ea 4c:<0d>
>01550 be 15 c9 ba f0 03 4c 6a:<36>
>01558 15 20 7d ff 1b 52 08 7b:<34>
>01560 1b 52 0b 00 ea ea ea 4c:<3e>
>01568 be 15 c9 b9 f0 03 4c 82:<f2>
>01570 15 20 7d ff 1b 52 08 7d:<53>
>01578 1b 52 0b 00 ea ea ea 4c:<6b>
>01580 be 15 c9 40 f0 03 4c 9a:<e3>
>01588 15 20 7d ff 1b 52 02 40:<01>
>01590 1b 52 0b 00 ea ea ea 4c:<9e>
>01598 be 15 c9 5c f0 03 4c b2:<83>
>015a0 15 20 7d ff 1b 52 03 5c:<f3>
>015a8 1b 52 0b 00 ea ea ea 4c:<4e>
>015b0 be 15 c9 a1 f0 03 4c be:<da>
>015b8 15 a9 dd 20 d2 ff c4 e7:<43>
>015c0 f0 04 c8 4c ac 13 a5 fb:<14>
>015c8 c9 2d d0 04 a5 fa c9 e0:<d3>
>015d0 d0 22 a9 0d 20 d2 ff 20:<e0>
>015d8 cc ff a9 04 20 c3 ff ad:<c3>
>015e0 5b 16 85 fa ad 5c 16 85:<41>
>015e8 fb ad 5d 16 85 fc ad 5e:<13>
>015f0 16 85 fd 60 18 a9 50 65:<eb>
>015f8 fa 85 fa a9 00 65 fb 85:<64>
>01600 fb 18 a9 50 65 fc 85 fc:<9b>
>01608 a9 00 65 fd 85 fd a9 0a:<73>
>01610 20 d2 ff 4c 95 13 a9 12:<37>
>01618 20 d2 ff 60 a9 92 20 d2:<b6>
>01620 ff 60 20 7d ff 1b 2d 01:<97>
>01628 00 60 20 7d ff 1b 2d 00:<28>
>01630 a9 00 20 d2 ff 60 20 1c:<c8>
>01638 16 20 2a 16 ad 5f 16 29:<fe>
>01640 01 c9 01 d0 04 20 16 16:<80>
>01648 60 ad 5f 16 29 02 c9 02:<94>
>01650 d0 03 20 22 16 60 00 00:<6b>
>01658 00 00 00 00 00 00 00 00:<c6>
```

```
>01660 bc bb bd be dc db dd b1:<a3>
>01668 ba b9 40 b0 ad 5c a5 a7:<69>
>01670 a3 a9 a8 aa a2 a4 ab a6:<27>
>01678 a1 5f ad 59 16 c9 01 d0:<06>
>01680 05 a9 1f 20 d2 ff 20 1c:<fd>
>01688 16 a5 fa 8d 5b 16 a5 fb:<f3>
>01690 8d 5c 16 a5 fc 8d 5d 16:<ae>
>01698 a5 fd 8d 5e 16 a9 60 85:<fc>
>016a0 fa a9 1c 85 fb a9 30 85:<72>
>016a8 fc a9 2e 85 fd ae 5a 16:<98>
>016b0 e0 00 f0 09 a9 20 20 d2:<de>
>016b8 ff ca 4c b0 16 20 33 17:<66>
>016c0 a0 00 a9 00 8d 5f 16 ad:<ef>
>016c8 5f 16 d1 fc f0 08 b1 fc:<80>
>016d0 8d 5f 16 20 8c 17 b1 fa:<2f>
>016d8 20 d2 ff c4 e7 f0 04 c8:<fe>
>016e0 4c c7 16 a5 fb c9 2d d0:<ea>
>016e8 04 a5 fa c9 e0 d0 22 a9:<18>
>016f0 0d 20 d2 ff 20 cc ff a9:<14>
>016f8 04 20 c3 ff ad 5b 16 85:<cc>
>01700 fa ad 5c 16 85 fb ad 5d:<5e>
>01708 16 85 fc ad 5e 16 85 fd:<e5>
>01710 60 18 a9 50 65 fa 85 fa:<a0>
>01718 a9 00 65 fb 85 fb 18 a9:<6f>
>01720 50 65 fc 85 fc a9 00 65:<ad>
>01728 fd 85 fd a9 0a 20 d2 ff:<3f>
>01730 4c ad 16 a0 00 b1 fc 29:<a3>
>01738 02 c9 02 f0 09 c4 e7 f0:<df>
>01740 04 c8 4c 35 17 60 a9 1f:<35>
>01748 20 d2 ff a0 00 b1 fc 29:<17>
>01750 02 c9 02 d0 08 a9 5f 20:<4f>
>01758 d2 ff 4c 62 17 a9 a0 20:<92>
>01760 d2 ff c4 e7 f0 04 c8 4c:<95>
>01768 4d 17 a9 8d 20 d2 ff ae:<44>
>01770 5a 16 e0 00 f0 09 a9 20:<17>
>01778 20 d2 ff ca 4c 72 17 ad:<c4>
>01780 59 16 c9 01 f0 05 a9 9f:<62>
>01788 20 d2 ff 60 20 1c 16 ad:<8c>
>01790 5f 16 29 01 c9 01 d0 03:<92>
>01798 20 16 16 60 15 33 33 33:<05>
```

SC 6

```
>01300 00 a9 60 85 fa a9 1c 85:<90>
>01308 fb a9 60 85 fc a9 1c 85:<bb>
>01310 fd ad 80 0f 85 eb ad 81:<95>
>01318 0f 85 ec 20 2d 1b a5 eb:<e2>
>01320 8d 85 0f a5 ec 8d 86 0f:<c6>
>01328 a9 1b 20 d2 ff 20 d2 ff:<39>
>01330 a5 e7 8d 8b 0f a9 13 20:<e7>
>01338 d2 ff 20 d2 ff ad 8b 0f:<52>
>01340 85 e7 a9 1d a2 04 20 d2:<49>
>01348 ff ca d0 fa a5 fc 8d 89:<f8>
>01350 0f a5 fd 8d 8a 0f 38 ad:<4b>
>01358 89 0f e9 60 8d 89 0f ad:<fd>
>01360 8a 0f e9 1c 8d 8a 0f a2:<af>
>01368 00 ad 8a 0f c9 00 d0 05:<de>
>01370 ad 89 0f c9 00 f0 15 38:<13>
>01378 ad 89 0f e9 50 8d 89 0f:<d5>
>01380 ad 8a 0f e9 00 8d 8a 0f:<13>
>01388 e8 4c 69 13 8a 20 fb f9:<aa>
>01390 a8 8a 20 d2 ff 98 20 d2:<fa>
>01398 ff a9 1d a2 06 20 d2 ff:<b0>
>013a0 ca d0 fa ad 86 0f 20 fb:<cb>
>013a8 f9 a8 8a 20 d2 ff 98 20:<c4>
>013b0 d2 ff a9 02 85 e5 a9 16:<21>
>013b8 85 e4 ad 85 0f 85 eb ad:<bb>
>013c0 86 0f 85 ec 20 2d 1b 20:<28>
>013c8 eb ee a0 00 d9 8c 0f d0:<a4>
>013d0 03 4c 0b 1a c8 c0 72 d0:<56>
>013d8 f3 a2 00 86 f4 c9 1d f0:<a0>
>013e0 03 4c 12 14 a5 ec c5 e7:<06>
>013e8 d0 03 4c f5 13 a9 1d 20:<31>
>013f0 d2 ff 4c 1e 13 a5 eb c9:<cb>
>013f8 16 d0 03 4c 0c 14 20 90:<b3>
>01400 1b a9 1d 20 d2 ff 4c 1e:<16>
>01408 13 4c 0f 14 4c ad 14 4c:<32>
>01410 41 18 c9 9d f0 03 4c 51:<dc>
>01418 14 a5 ec c9 00 d0 03 4c:<78>
>01420 2a 14 a9 9d 20 d2 ff 4c:<4e>
>01428 1e 13 a5 eb c9 02 d0 03:<d4>
>01430 4c 41 14 20 9e 1b a9 9d:<98>
>01438 20 d2 ff 4c 1e 13 4c 4e:<04>
>01440 14 20 00 19 a9 11 20 d2:<8f>
>01448 ff a9 9d 20 d2 ff 4c 41:<19>
>01450 18 c9 13 f0 03 4c 6f 14:<69>
>01458 a5 fa 85 fc a5 fb 85 fd:<e9>
>01460 a9 13 20 d2 ff a9 00 85:<fa>
>01468 f0 4c 1e 13 4c 41 18 c9:<d5>
>01470 93 f0 03 4c 8f 14 20 d2:<32>
>01478 ff a9 60 85 fa a9 1c 85:<7e>
>01480 fb a5 fa 85 fc a5 fb 85:<43>
>01488 fd 20 44 18 4c 41 18 c9:<8b>
>01490 1b f0 03 4c a6 14 a5 eb:<19>
>01498 8d 80 0f a5 ec 8d 81 0f:<8c>
>014a0 a9 01 60 4c 41 18 c9 0d:<b0>
>014a8 f0 03 4c e3 14 20 90 1b:<56>
>014b0 a5 eb c9 16 f0 03 4c db:<a3>
>014b8 14 a5 fd c9 2e d0 04 a5:<cf>
>014c0 fc c9 30 d0 06 20 9e 1b:<bd>
>014c8 4c d1 14 20 9e 1b 20 64:<4c>
>014d0 19 a9 00 85 ec 20 2d 1b:<e7>
>014d8 4c e0 14 a9 0d 20 d2 ff:<70>
>014e0 4c 41 18 c9 11 f0 03 4c:<40>
>014e8 07 15 a5 eb c9 16 d0 03:<c5>
>014f0 4c 01 15 a9 11 20 d2 ff:<e6>
>014f8 20 90 1b 4c 1e 13 4c 04:<64>
>01500 15 4c da 15 4c 41 18 c9:<8a>
>01508 91 f0 03 4c 2b 15 a5 eb:<f8>
>01510 c9 02 d0 03 4c 25 15 a9:<60>
>01518 91 20 d2 ff 20 9e 1b 4c:<f8>
>01520 1e 13 4c 28 15 4c be 15:<dc>
>01528 4c 41 18 c9 8b f0 03 4c:<e7>
>01530 69 15 a4 ec c4 e7 f0 1c:<e1>
>01538 b1 fc c9 20 f0 08 c4 e7:<37>
```

```
>01540 f0 12 c8 4c 38 15 b1 fc:<97>
>01548 c9 20 d0 12 c4 e7 f0 04:<a5>
>01550 c8 4c 46 15 a5 e7 85 ec:<14>
>01558 20 2d 1b 4c 1e 13 84 ec:<19>
>01560 20 2d 1b 4c 1e 13 4c 41:<d2>
>01568 18 c9 89 f0 03 4c a8 15:<3f>
>01570 a4 ec c0 00 f0 1c 88 b1:<6d>
>01578 fc c9 20 d0 07 c0 00 f0:<72>
>01580 11 4c 76 15 88 b1 fc c9:<c4>
>01588 20 f0 11 c0 00 f0 03 4c:<4d>
>01590 84 15 a9 00 85 ec 20 2d:<fb>
>01598 1b 4c 1e 13 c8 84 ec 20:<12>
>015a0 2d 1b 4c 1e 13 4c 41 18:<f2>
>015a8 c9 a0 f0 03 4c b7 15 a9:<7b>
>015b0 20 4c 0b 1a 4c 41 18 c9:<83>
>015b8 85 f0 03 4c d3 15 a5 fd:<19>
>015c0 c9 1c d0 04 a5 fc c9 60:<63>
>015c8 d0 03 4c 1e 13 20 00 19:<60>
>015d0 4c 41 18 c9 87 f0 03 4c:<79>
>015d8 ef 15 a5 fd c9 2d d0 04:<b2>
>015e0 a5 fc c9 e0 d0 03 4c 1e:<15>
>015e8 13 20 64 19 4c 41 18 c9:<e7>
>015f0 09 f0 03 4c fc 15 20 d2:<a1>
>015f8 ff 4c 41 18 c9 0a f0 03:<ae>
>01600 4c 09 16 4c ea 14 4c 41:<c3>
>01608 18 c9 8d f0 03 4c 9a 16:<90>
>01610 a5 fd c9 2d d0 04 a5 fc:<0a>
>01618 c9 e0 d0 03 4c 1e 13 a5:<a5>
>01620 eb c9 16 d0 0b 20 64 19:<fc>
>01628 20 9e 1b a9 91 20 d2 ff:<74>
>01630 20 c4 18 c0 01 f0 08 a9:<e0>
>01638 07 20 d2 ff 4c 1e 13 a4:<8c>
>01640 ec a2 00 b1 fc 9d 20 10:<d6>
>01648 a9 20 91 fc 20 66 1b b1:<e4>
>01650 fc 9d 72 10 a9 00 91 fc:<de>
>01658 20 74 1b c4 e7 f0 05 c8:<f3>
>01660 e8 4c 43 16 e8 a9 00 9d:<17>
>01668 20 10 20 90 1b a9 11 20:<47>
>01670 d2 ff 20 06 0b a0 00 b9:<70>
>01678 20 10 c9 00 f0 11 91 fc:<ec>
>01680 20 66 1b b9 72 10 91 fc:<b1>
>01688 20 74 1b c8 4c 77 16 20:<fb>
>01690 44 18 a9 4a 20 1e c0 4c:<7c>
>01698 41 18 c9 14 f0 03 4c dc:<98>
>016a0 17 a4 ec c0 00 f0 30 b1:<cd>
>016a8 fc 88 91 fc c8 20 66 1b:<0c>
>016b0 b1 fc 88 91 fc 20 74 1b:<b1>
>016b8 c4 e7 f0 05 c8 c8 4c a7:<61>
>016c0 16 a9 20 91 fc 20 66 1b:<d6>
>016c8 a9 00 91 fc 20 74 1b a9:<5e>
>016d0 14 20 d2 ff 4c 1e 13 a5:<ea>
>016d8 fd c9 1c d0 04 a5 fc c9:<95>
>016e0 60 d0 03 4c 1e 13 a5 fd:<53>
>016e8 c5 fb d0 04 a5 fc c5 fa:<06>
>016f0 d0 0b 20 00 19 20 90 1b:<25>
>016f8 a9 11 20 d2 ff 20 9e 1b:<f2>
>01700 a4 e7 a2 00 b1 fc c9 20:<dc>
>01708 d0 09 c0 00 f0 0f 88 e8:<ff>
>01710 4c 04 17 c4 e7 d0 03 4c:<55>
>01718 d4 17 c8 c8 ca 20 90 1b:<01>
>01720 8c 87 0f 8a a8 c0 00 d0:<dd>
>01728 03 4c d1 17 b1 fc c9 20:<cb>
>01730 f0 04 88 4c 25 17 8c 84:<b9>
>01738 0f a9 00 99 20 10 88 b1:<6b>
>01740 fc 99 20 10 20 66 1b b1:<14>
>01748 fc 99 72 10 20 74 1b c0:<52>
>01750 00 f0 04 88 4c 3f 17 ac:<2c>
>01758 84 0f c4 e7 f0 26 c8 a2:<36>
>01760 00 b1 fc 20 ba 1b 91 fc:<8d>
>01768 20 66 1b 20 ba 1b b1 fc:<6b>
>01770 20 ba 1b 91 fc 20 ba 1b:<f9>
>01778 20 74 1b c4 e7 f0 07 c8:<31>
>01780 e8 4c 61 17 a2 ff 20 ba:<2c>
>01788 1b c8 a9 20 91 fc 20 66:<4e>
>01790 1b a9 00 91 fc 20 74 1b:<97>
>01798 c4 e7 f0 04 c8 4c 8a 17:<ee>
>017a0 ac 87 0f a2 00 20 9e 1b:<5d>
>017a8 bd 20 10 c9 00 f0 12 91:<b6>
>017b0 fc 20 66 1b bd 72 10 91:<99>
>017b8 fc 20 74 1b e8 c8 4c a8:<46>
>017c0 17 ac 87 0f 84 ec c6 eb:<78>
>017c8 20 2d 1b 20 44 18 4c 1e:<a9>
>017d0 13 20 9e 1b a9 9d 20 d2:<86>
>017d8 ff 4c 41 18 c9 94 f0 03:<c2>
>017e0 4c 29 18 4c c7 1b 4c 3e:<fb>
>017e8 18 a4 e7 b1 fc c9 20 f0:<4a>
>017f0 03 4c e5 1b a9 94 20 d2:<0d>
>017f8 ff a9 00 85 f5 88 b1 fc:<2f>
>01800 c8 91 fc 88 20 66 1b b1:<7d>
>01808 fc c8 91 fc 20 74 1b 88:<51>
>01810 c4 ec f0 03 4c fd 17 a9:<2a>
>01818 20 91 fc 20 66 1b a9 00:<ae>
>01820 91 fc 20 74 1b 60 4c 41:<36>
>01828 18 a5 d3 29 04 c9 04 d0:<a0>
>01830 0d a5 eb 8d 80 0f a5 ec:<df>
>01838 8d 81 0f a9 02 60 4c c7:<9b>
>01840 13 4c 1e 13 a5 eb 8d 85:<21>
>01848 0f a5 ec 8d 86 0f a9 13:<04>
>01850 20 d2 ff a9 1b 20 d2 ff:<6f>
>01858 20 d2 ff a2 00 8e 87 0f:<b0>
>01860 a0 00 20 4a 1b b1 fa cd:<f4>
>01868 87 0f f0 06 8d 87 0f 20:<eb>
>01870 9e 18 20 58 1b b1 fa 20:<f8>
>01878 d2 ff c4 e7 f0 04 c8 4c:<c5>
>01880 62 18 e0 14 f0 07 e8 20:<b5>
>01888 82 1b 4c 60 18 20 ac 1b:<0b>
>01890 ad 85 0f 85 eb ad 86 0f:<dd>
>01898 85 ec 20 2d 1b 60 a9 1b:<fd>
>018a0 20 d2 ff 20 d2 ff ad 87:<85>
>018a8 0f 29 01 c9 01 d0 06 a9:<fc>
>018b0 12 20 d2 ff 60 ad 87 0f:<af>
>018b8 29 02 c9 02 d0 05 a9 02:<f2>
>018c0 20 d2 ff 60 a5 fa 8d 85:<0d>
>018c8 0f a5 fb 8d 86 0f a9 e0:<b8>
>018d0 85 fa a9 2d 85 fb a0 00:<de>
>018d8 b1 fa c9 20 d0 08 c4 e7:<f5>
```

```
>018e0 f0 11 c8 4c d8 18 a0 00:<d9>
>018e8 ad 85 0f 85 fa ad 86 0f:<28>
>018f0 85 fb 60 a0 01 ad 85 0f:<29>
>018f8 85 fa ad 86 0f 85 fb 60:<d9>
>01900 a5 fb c9 1c d0 04 a5 fa:<9b>
>01908 c9 60 d0 01 60 38 a5 fa:<2f>
>01910 e9 50 85 fa a5 fb e9 00:<9c>
>01918 85 fb 20 9e 1b a9 57 20:<de>
>01920 1e c0 a5 eb 8d 85 0f a5:<64>
>01928 ec 8d 86 0f a9 13 20 d2:<da>
>01930 ff a0 00 8c 87 0f 20 4a:<13>
>01938 1b b1 fa cd 87 0f f0 06:<9d>
>01940 8d 87 0f 20 9e 18 20 58:<92>
>01948 1b b1 fa 20 d2 ff c4 e7:<74>
>01950 f0 04 c8 4c 36 19 ad 85:<a3>
>01958 0f 85 eb ad 86 0f 85 ec:<56>
>01960 20 2d 1b 60 a5 fb c9 27:<33>
>01968 d0 04 a5 fa c9 a0 d0 01:<29>
>01970 60 20 82 1b 20 90 1b a9:<8a>
>01978 56 20 1e c0 a5 eb 8d 85:<e3>
>01980 0f a5 ec 8d 86 0f a9 16:<0d>
>01988 85 eb a9 00 85 ec 20 2d:<9c>
>01990 1b 18 a9 40 65 fa 85 fa:<dd>
>01998 a9 06 65 fb 85 fb a0 00:<4c>
>019a0 8c 87 0f 20 4a 1b b1 fa:<63>
>019a8 cd 87 0f f0 06 8d 87 0f:<1b>
>019b0 20 9e 18 20 58 1b b1 fa:<49>
>019b8 20 d2 ff c4 e7 f0 04 c8:<3a>
>019c0 4c a3 19 ad 85 0f 85 eb:<06>
>019c8 ad 86 0f 85 ec 20 2d 1b:<20>
>019d0 20 ac 1b 60 a4 ec 8c 86:<9d>
>019d8 0f c4 e7 f0 08 ad 83 0f:<fc>
>019e0 f0 03 4c d4 1b ac 86 0f:<d0>
>019e8 ad 84 0f 91 fc ad 00 13:<48>
>019f0 8d 87 0f 20 9e 18 ad 84:<9e>
>019f8 0f 20 d2 ff 20 66 1b ad:<04>
>01a00 00 13 ac 86 0f 91 fc 20:<51>
>01a08 74 1b 60 a4 ec 8d 84 0f:<2b>
>01a10 c4 e7 f0 0c 4c eb 1b a2:<97>
>01a18 00 86 f4 86 f0 4c 1e 13:<d2>
>01a20 20 c4 18 c0 01 f0 10 4c:<19>
>01a28 fe 1b a9 07 20 d2 ff a9:<c9>
>01a30 9d 20 d2 ff 4c ad 14 a5:<4c>
>01a38 fd c9 2d d0 04 a5 fc c9:<05>
>01a40 e0 d0 0b 4c 11 1c a9 07:<eb>
>01a48 20 d2 ff 4c 1e 13 a5 eb:<af>
>01a50 c9 16 d0 0b 20 64 19 a9:<54>
>01a58 91 20 d2 ff 20 9e 1b 20:<87>
>01a60 90 1b a9 11 20 d2 ff 20:<d8>
>01a68 06 0b a9 91 20 d2 ff 20:<3a>
>01a70 9e 1b ad 84 0f a2 00 dd:<51>
>01a78 78 0f f0 08 e8 e0 08 d0:<95>
>01a80 f6 4c 8d 1a 4c 24 1c 20:<11>
>01a88 90 1b 4c 1e 13 a4 e7 88:<d3>
>01a90 b1 fc c9 20 f0 07 c0 00:<dc>
>01a98 f0 ea 4c 8f 1a a4 e7 a9:<d4>
>01aa0 20 91 fc 20 66 1b a9 00:<7b>
>01aa8 91 fc 20 74 1b a2 00 88:<ba>
>01ab0 b1 fc c9 20 f0 1b 9d 20:<a1>
>01ab8 10 a9 20 91 fc 20 66 1b:<dc>
>01ac0 b1 fc 9d 72 10 a9 00 91:<49>
>01ac8 fc 20 74 1b e8 88 4c b0:<e8>
>01ad0 1a c8 ca 20 ba 1b 86 ec:<e5>
>01ad8 20 2d 1b a9 51 20 1e c0:<76>
>01ae0 20 90 1b a9 4a 20 1e c0:<9c>
>01ae8 a9 11 20 d2 ff ac 86 0f:<14>
>01af0 a2 00 c0 ff f0 31 20 66:<74>
>01af8 1b bd 72 10 91 fc 8d 87:<a5>
>01b00 0f 20 74 1b bd 20 10 91:<a9>
>01b08 fc 20 9e 18 bd 20 10 84:<b4>
>01b10 ec 20 2d 1b 20 d2 ff c0:<cc>
>01b18 00 f0 05 88 e8 4c f6 1a:<0e>
>01b20 c6 ec a9 4b 20 1e c0 4c:<8d>
>01b28 37 1c 4c 1e 13 8d 89 0f:<5f>
>01b30 8e 8a 0f 8c 8b 0f a6 eb:<1e>
>01b38 a4 ec ca ca 18 20 18 c0:<68>
>01b40 ae 8a 0f ac 8b 0f ad 89:<70>
>01b48 0f 60 18 a9 d0 65 fa 85:<07>
>01b50 fa a9 11 65 fb 85 fb 60:<c7>
>01b58 38 a5 fa e9 d0 85 fa a5:<da>
>01b60 fb e9 11 85 fb 60 18 a9:<0f>
>01b68 d0 65 fc 85 fc a9 11 65:<04>
>01b70 fd 85 fd 60 38 a5 fc e9:<7e>
>01b78 d0 85 fc a5 fd e9 11 85:<ef>
>01b80 fd 60 18 a9 50 65 fa 85:<52>
>01b88 fa a9 00 65 fb 85 fb 60:<d3>
>01b90 18 a9 50 65 fc 85 fc a9:<bb>
>01b98 00 65 fd 85 fd 60 38 a5:<73>
>01ba0 fc e9 50 85 fc a5 fd e9:<97>
>01ba8 00 85 fd 60 38 a5 fa e9:<71>
>01bb0 40 85 fa a5 fb e9 06 85:<43>
>01bb8 fb 60 8c 85 0f 8e 86 0f:<72>
>01bc0 ae 85 0f ac 86 0f 60 20:<21>
>01bc8 e9 17 c9 00 d0 03 4c 1e:<26>
>01bd0 13 4c e6 17 a9 01 8d 4a:<b3>
>01bd8 1c 20 e9 17 ad 4a 1c f0:<a0>
>01be0 03 4c e5 19 60 a9 00 8d:<56>
>01be8 4a 1c 60 20 4b 1c 20 d4:<f7>
>01bf0 19 ad 4a 1c c9 00 d0 03:<87>
>01bf8 4c 1e 13 4c 17 1a 20 4b:<9d>
>01c00 1c 20 d4 19 ad 4a 1c c9:<22>
>01c08 00 d0 03 4c 1e 13 4c 2a:<a6>
>01c10 1a 20 4b 1c 20 d4 19 ad:<bd>
>01c18 4a 1c c9 00 d0 03 4c 1e:<71>
>01c20 13 4c 46 1a 20 4b 1c 20:<70>
>01c28 d4 19 ad 4a 1c c9 00 d0:<32>
>01c30 03 4c 1e 13 4c 87 1a 20:<d4>
>01c38 4b 1c 20 d4 19 ad 4a 1c:<f6>
>01c40 c9 00 d0 03 4c 1e 13 4c:<24>
>01c48 2a 1b 00 a9 01 8d 4a 1c:<74>
>01c50 60 00 00 00 00 00 00 00:<1c>
```

SC 7

```
>03620 34 20 0d 20 32 20 0d 20:<09>
>03628 34 20 0d 20 37 20 0d 20:<ea>
```

```
>03630 37 20 0d 20 30 20 0d 20:<8d>
>03638 30 20 0d 00 00 00 00 00:<4b>
```

Installation Epson

```
1 rem installationsprogramm epson   <hp>
2 rem zu scriptcall by martin scho
lz                                  <go>
10 fori=0to39:cd$=cd$+chr$(17):c1$
=c1$+chr$(157):next                 <oc>
11 cy$=chr$(159)                    <no>
100 graphic5,1:fast:color6,7:pudef
"0"                                 <pi>
110 printcy$"      **** scriptcall
- installationsprogramm fuer epson
 drucker ***"                       <kj>
120 printleft$(cd$,5)"orginaldiske
tte einlegen <cr>"                  <di>
130 getkeya$:ifa$<>chr$(13)then130  <ee>
140 bload"sc 5"                     <dm>
150 printleft$(cd$,3):restore:w=10  <db>
160 input"schriftart waehlen, wiev
iele werte max 5:";w                <oc>
170 ifw>5thenw=5:elseifw<1thenw=1   <np>
180 fort=1tow:printt". wert:  ";:p
rintusing"###";peek(4943+t-1);:pri
ntleft$(c1$,5);:inpute:poke4943+t-
1,e:next:poke4943+t-1,0:w=w+1       <on>
190 ifw<6thenw=w+1:poke4943+w-1,23
4:goto190                           <ki>
200 printchr$(17);:input"schoensch
rift einschalten max 5 werte:";w    <nd>
210 ifw>5thenw=5:elseifw<1thenw=1   <aa>
220 fort=1tow:printt". wert:  ";:p
rintusing"###";peek(4965+t-1);:pri
ntleft$(c1$,5);:inpute:poke4965+t-
1,e:next:poke4965+t-1,0:w=w+1       <ja>
230 ifw<6thenw=w+1:poke4965+w-1,23
4:goto230                           <jh>
240 foru=1to12:readz$,z             <gk>
250 printchr$(17)z$" max 10 werte:
";:inputw                           <ed>
260 ifw>10thenw=10:elseifw<1thenw=
1                                   <gc>
270 fort=1tow:printt". wert:  ";:p
rintusing"###";peek(z+t-1);:printl
eft$(c1$,5);:inpute:pokez+t-1,e:ne
xt:pokez+t-1,0:w=w+1                <be>
280 ifw<11thenw=w+1:pokez+w-1,234:
goto280                             <nc>
290 next                            <mi>
300 printleft$(cd$,2)"alles richti
g (j/n) ?"                          <ci>
310 getkeya$:ifa$<>"j"anda$<>"n"th
en310                               <bl>
320 ifa$<>"j"then150                <ej>
330 printleft$(cd$,5)"orginaldiske
tte einlegen <cr>"                  <lm>
340 getkeya$:ifa$<>chr$(13)then340  <cp>
350 scratch"sc 5":bsave"sc 5",b0,p
4864top6044                         <fb>
360 end                             <fc>
370 data kleines ae,5276            <bd>
380 data kleines oe,5300            <gi>
390 data kleines ue,5324            <oi>
400 data sz,5348                    <ej>
410 data grosses ae,5372            <fg>
420 data grosses oe,5396            <ok>
430 data grosses ue,5420            <il>
440 data grad-symbol,5444           <nh>
450 data geschweifte klammer auf,5
468                                 <df>
460 data geschweifte klammer zu,54
92                                  <pc>
470 data paragraph-symbol,5516      <ko>
480 data querstrich,5540            <fo>
```

Script-Call

Fortsetzung von Seite 56

LINEFEED	– führt einen Zeilenvorschub ohne Return aus.
TAB	– springt den nächsten Tabulator an.
F1/F3	– scrollt die Seite hoch und runter.
F5/F7	– springt zum letzten beziehungsweise nächsten Wortanfang.
ESC	– verläßt den WRITE-Modus und springt in den FUNKTION-Modus (siehe unten).

Zusätzlich enthält das Programm einen automatischen Zeilenumbruch, der die Eingabe von Texten wesentlich erleichter, das sogenannte Word-Wrapping.

COMMAND-Modus:
Dieser Modus, den man durch Drücken von CONTROL erreicht, stellt folgende Zusatzfunktionen durch einfaches Drücken der angegebenen Taste zur Verfügung:

Q / P	– löscht die Zeile ab der / bis zur Cursorposition.
I / D	– löscht Zeile / fügt Zeile ein.
@ / ↑	– löscht Seite ab / bis Cursor.
L / R	– positioniert die Zeile rechts / links vom Cursor.
J / K	– springt zum Anfang / Ende der Zeile.
Z / Y	– löscht Tabulatoren / setzt diese auf Standard.
U / 9	– schaltet unterstreicht / revers-Modus ein / aus.
M	– zentriert die Zeile.
A	– schaltet Insertmodus ein / aus.
*	– löscht die ganze Seite.
TAB	– setzt / löscht Tabulator.
CONTROL	– schaltet zurück in den WRITE-Modus.

FUNKTION-Modus:
Dieser Modus, den man durch Drücken von ESC erreicht, ist der Menü-Modus mit den Funktionen zur Verwaltung der Texte. Hier haben folgende Tasten die ihnen zugeordneten Bedeutungen:

ESC	– springt zurück in den WRITE-Modus.
CRSR-Tasten	– wählen einen Menüpunkt aus.

RETURN	– wählt diesen an.
HELP	– zeigt eine Hilfsseite mit sämtlichen Editierkommandos.

DAS HAUPTMENÜ ENTHÄLT FOLGENDE PUNKTE:

Page:
Hier können Sie eine der fünf Seiten anwählen, die Sie editieren wollen. Der Punkt RETURN ermöglicht das vorzeitige Verlassen dieses Untermenüs.

Search:
Dieser Punkt bietet die Möglichkeit, die aktuelle Seite nach frei wählbaren Texten zu durchsuchen. Wurde der von Ihnen eingegebene Text gefunden, so erscheint an der betreffenden Stelle ein schnell blinkender Blockcursor. Wird nun RETURN gedrückt, so springt das Programm in den WRITE-Modus und die gefundene Stelle kann editiert werden. Drücken Sie an gleicher Stelle die SPACE-Taste, so wird das Suchen auf dieser Seite fortgesetzt.

Block:
Dieser Menüpunkt enthält folgende Blockoperationen:

Cut	– schneidet einen frei wählbaren Bereich von bis zu zwölf Zeilen aus der aktuellen Seite aus und kopiert diesen in einen unabhängigen Zwischenspeicher. Im Text wird dieser Bereich gelöscht.
Copy	– erledigt das gleiche, nur wird der ausgewählte Abschnitt im Text nicht gelöscht, sondern er bleibt erhalten.
Paste	– ermöglicht es nun, den Block an jeder beliebigen Stelle auf jeder beliebigen Seite einzufügen.

Delete:
Dieser Menüpunkt dient zum Löschen einer bestimmten oder aller Seiten.
Achtung: Sie sind unwiederbringlich verloren.

Disk:
Dieses Menü bietet folgende diskettenbezogene Punkte:

Directory	– Ausgabe des Directorys der eingelegten Diskette.
Format Disk	– Formatieren einer Datendiskette. Nach einer Kontrollabfrage wird die eingelegte Diskette hart formatiert.
Load Text	– Laden eines Textes.
Save Text	– Speichern des Textes. Ist bereits ein Text unter gleichem Namen auf Diskette vorhanden, so wird gefragt, ob dieser überschrieben werden soll.
Save Parameter	– Sämtliche im Programm einstellbaren Parameter werden auf Diskette gespeichert. Bei jedem späteren Arbeiten mit Scriptcall sind diese Werte dann automatisch voreingestellt.

Selbstverständlich werden auch hier alle auftretenden Fehler abgefangen und angezeigt.

Print:
Dieser Menüpunkt dient zum Ausdrucken der Texte. Zuvor stellt man in zwei aufeinanderfolgenden Menüs die Anfangsseite (First page) sowie die Endseite (Last page), den Randvorschub (Left margin), die Druckschriftart (Print style – nur für Epson-Drucker) und den möglichen Schönschrift-Modus (NLQ) ein. Nachdem man im ersten Fenster den Punkt GO ON anwählte, beginnt nach Anwählen des Punktes PRINT im zweiten Fenster der Ausdruck wahlweise auf einem Epson-kompatiblen Drucker, oder dem Standarddrucker zum PC128, dem MPS 1000, je nach Einstellung im Setup-Menü. Ferner können dort auch Geräte- und Sekundäradresse eingestellt werden.

Setupt:
Hier können die für das Programm wichtigen Einstellungen vorgenommen werden:

Columns	– Einstellen der Textbreite. Wird dieser Punkt ordnungsgemäß mit RETURN abgeschlossen, so wird der augenblickliche Text unwiederbringlich gelöscht, da intern der Speicher neu organisiert werden muß. Wählt man dagegen ABORT, so bleibt der Text, aber auch die alte Einstellung der Textbreite erhalten.
Printer device	– Die Geräte- und Sekundäradresse des Druckers kann eingestellt werden.
Printer Type	– Einstellen der unterschiedlichen Druckertypen (MPS 1000 oder Epson-kompatible Drucker). Wählt man letzteren, so ist mittels des separaten Installationsprogrammes die Druckroutine an den jeweiligen Drucker anzupassen. Dazu später mehr.

Special:
Dieses Menü enthält folgende zwei Punkte:

Reset	– versetzt den Computer in den Einschaltzustand.
Restart	– führt einen Neustart des Programms durch. Ein eventuell im Speicher vorhandener Text wird gelöscht.

Wie bereits erwähnt, muß ein Epson-kompatibler Drucker an den Druckertreiber angepaßt werden. Zu diesem Zweck befindet sich auf der Diskette ein Installationsprogramm unter dem Namen INSTALL EPSON, welches zu laden und zu starten ist. Hier müssen die für den Drucker gültigen ESCape-Sequenzen eingegeben werden. Als ersten Punkt findet man SCHRIFTART WÄHLEN vor. Die meisten Drucker erlauben es, mittels nur einer ESCape-Sequenz zwischen den verschiedenen Schriftarten zu wählen. Besitzt Ihr Drucker diese Möglichkeit, so geben Sie zunächst an, wieviele Codes dazu nötig sind. Die daraufhin einzugebenden Codes werden dann vor jedem Ausdruck mit Scriptcall, gefolgt von dem im PRINT-Menü unter dem Punkt PRINT STYLE eingestellten Wert, an den Drucker gesendet.
Als nächstes muß man die ESCape-Sequenz zum Anschalten der NLQ-Schrift eingeben, und dann fragt das Programm mehrere Codes ab, die es zur korrekten Ausgabe der Sonderzeichen (öäüÖÄÜß*()\) an den Drucker senden soll. Wurden alle Werte korrekt eingestellt, so schreibt das Programm den korrekten Druckertreiber auf Diskette zurück, und der Ausdruck mit Scriptcall kann beginnen. *Martin Scholz* □

HEIZKOSTEN-ABRECHNUNG

Wenn der Gasmann zwei mal klingelt

Auch dieses Jahr wird Ihnen wieder die Heizkosten-Rechnung von Ihrem Vermieter oder der Hausverwaltung präsentiert werden. Aber ist sie auch in Ordnung?

Für die meisten Mieter ist eine vom Computer erstellte Heizkosten-Abrechnung ein Buch mit sieben Siegeln, wird als gegeben hingenommen und ohne Kommentar bezahlt.
Das muß nicht sein. Dies zeigt unser Listing für den 128PC im 80-Zeichen-Modus. Es überprüft unbestechlich die Angaben des Rechnungsstellers auf Richtigkeit.

BEZAHLEN IST GUT – KONTROLLE BESSER

Allerdings – eins kann das Programm nicht: die internen Angaben des Vermieters überprüfen. Dafür muß der Mieter selbst sorgen, indem er Einsicht in die Abrechnungsunterlagen nimmt. Die Möglichkeit dazu ist gesetzlich verankert.
Haben Sie sich von der Richtigkeit überzeugt, errechnet das Programm die der Abrechnung zugrunde liegenden Angaben.
Geben Sie die aus dem Abrechnungsformular ersichtlichen Daten ein und berücksichtigen Sie dabei auch eventuell geleistete Vorauszahlungen, so wird eine in Frage kommende Gutschrift oder Nachzahlung ermittelt.

AUTOR BIETET ANPASSUNG AN

Außerdem ist im Programm eine kleine Information eingebaut, die auf die Rechtsgrundlage sowie die für die Abrechnung geltende Formel hinweist.
Der Autor erklärt sich beispielgebend für alle anderen bereit, das Programm an besondere Abrechnungs-Modalitäten, wie andere prozentuale Verteilung, des Benutzers anzupassen.
Schicken Sie dazu lediglich eine Kopie der Originalabrechnung mit Leerdisk und Freiumschlag an:
Günter Kl.-übbing, Eschenweg 8, 4290 Bocholt. □

Heizkostenabrechnung

```
10 rem ======================128   <ob>
20 rem (p) commodore welt       ==   <hf>
30 rem ==========================    <mm>
40 rem (c) by                   ==   <pp>
50 rem      guenter kl.-uebbing ==   <ph>
60 rem                          ==   <nd>
70 rem version 7.0   80z./ascii==    <gi>
80 rem pc-128 floppy/datasette ==    <ho>
90 rem ==========================    <km>
95 gosub 60000                       <jp>
100 printchr$(11);chr$(14):fast      <bb>
110 te$="HEIZKOSTENABRECHNUNG"       <nc>
120 ch$=chr$(145)                    <lg>
130 x=0.01:rem*rundungsfaktor        <bf>
140 ro$=chr$(18)                     <oj>
150 rf$=chr$(146)                    <ck>
160 be$=chr$(15)                     <gj>
170 ba$=chr$(143)                    <op>
180 u$=chr$(2)                       <md>
190 pudef" .,"                       <oj>
200 gosub3010                        <mk>
210 print                            <do>
220 getkeyx$:t=val(x$):ift<0ort>3t
hen220                               <da>
230 ontgosub250,2340,2940            <ek>
240 goto210                          <kk>
250 rem*berechn.brennstoff           <jl>
260 printchr$(27)+"u";chr$(27)+"m"   <dj>
270 scnclr:char0,23,0,te$:print:pr
int:printchr$(2);:print"A.) Berech
nung Brennstoffverbrauch + Kosten"   <ge>
280 print:print                      <dm>
290 gh=gh+gh                         <dc>
300 input"1. Brennstoffverbrauch l
t. Rechnungen des Ausstellers    :
     ";bv                            <mc>
310 printspc(62);ch$;:printusing"#
#####.># KW";bv                      <ic>
320 print                            <bl>
330 input"2. Brennstoffkosten lt.
Rechnungen des Ausstellers       :
     ";bk                            <af>
340 printspc(62);ch$;:printusing"#
#####.># DM";bk                      <fn>
350 print                            <fh>
360 input"3. Heiznebenkosten Gebue
hr Verbrauchserfassung pp.       :
     ";hn                            <ed>
370 printspc(62);ch$;:printusing"#
#####.># DM";hn                      <ia>
380 print:kh=bk+hn                   <me>
390 print"4. Die Kosten der Heizan
lage betragen                    :
   ";:printusing"#######.># DM";kh   <fc>
400 print                            <lm>
410 input"5. Zusatzkosten Wartung
pp.                              :
```

```
     ";zk                                    <bp>
420 printspc(62);ch$;:printusing"#
#####.># DM";zk                          <af>
430 gh=bk+hn+zk:print                     <jn>
440 print"6. Die Gesamtkosten der
Heizanlage betragen           :
   ";ro$;:printusing"#######.># DM
";gh                                      <ka>
450 gosub3160                             <dk>
460 getkeyx$                              <ji>
470 ifx$="w"then530                       <km>
480 ifx$="z"then260                       <pg>
490 ifx$="m"then200                       <pp>
500 ifx$="h"thengosub2260:goto450         <ie>
510 goto460                               <lb>
520 rem*berechn. heizwert warmwass
er                                        <jf>
530 scnclr:char0,23,0,te$:print:pr
int:printchr$(2);:print"B.) Berech
nung Heizwert Warmwasser"                 <ok>
540 print:print                           <ge>
550 print"1. Der Brennstoffverbrau
ch aus A - Ziffer 1 betraegt   :
   ";:printusing"#######.># KW";bv <ke>
560 print                                 <pn>
570 input"2. Gesamtverbrauch Warmw
asser lt. Rechnungssteller     :
     ";vb                                 <gg>
580 printspc(62);ch$;:printusing"#
#####.># KW";vb                           <ac>
590 print                                 <dj>
600 kwh=2.5*vb*(60-10)/1.00               <dp>
610 print"3. Ergibt nach Paragraph
 9 der Heizkostenverordnung"              <hn>
620 print"   vom 05.04.1984 einen
Heizwert von        >>>        :
   ";:printusing"#######.># KW";kw
h                                         <pk>
630 print                                 <ij>
640 ps=kwh*100/bv                         <cl>
650 print"4. Der Wert aus B 3.) di
vidiert durch Brennstoffver-"             <io>
660 print"   brauch  B 1.) ergibt
den prozentualen Verbrauch"               <ln>
670 print"   der Heizanlage von
                 >>>           :
        ";:printusing"##.># %";ps  <jg>
680 gosub3160                             <ce>
690 getkeyx$                              <ai>
700 ifx$="w"then760                       <pb>
710 ifx$="z"then530                       <le>
720 ifx$="m"then200                       <lb>
730 ifx$="h"thengosub2260:goto680         <ah>
740 goto690                               <cn>
750 rem*brechn. warmwasserkosten          <lb>
760 scnclr:char0,23,0,te$:print:pr
int:printchr$(2);:print"C.) Berech
nung Warmwasserkosten + Nebenkoste
n"                                        <cn>
770 print:print                           <jm>
780 print"1. Die Kosten der Heizan
lage aus A 4.)  betragen         :
   ";:printusing"#######.># DM";kh <ao>
790 print                                 <ml>
800 print"2. Der nach Formel ermit
telte prozentuale Heizwert ist :
        ";:printusing"##.># %";ps  <cm>
810 print                                 <pd>
820 input"3. % -Wert bei Bedarf au
f 1 Stelle hinter Komma runden  :
     ";ps                                 <nn>
830 printspc(62);ch$;:printusing"#
#####.>  %";ps                            <bc>
840 print:gwk=ps*kh/100                   <jh>
850 print"4. Somit ergeben sich Wa
rmwasserkosten "ps" % v. B 1):
 ";ro$;:printusing"#######.># DM";
gwk                                       <pd>
860 print                                 <fh>
870 print"5. Zuezueglich Nebenkost
en aus A 5.)                    :
   ";:printusing"#######.># DM";zk <ha>
880 print:hz=gwk+zk                       <jp>
890 print"6. Die Gesamtwarmwasserk
osten betragen somit ( 4 + 5 ) :
   ";:printusing"#######.># DM";hz <in>
900 pg=hz*40/100:wp=hz*60/100             <bg>
910 gosub3160                             <af>
920 getkeyx$                              <hp>
930 ifx$="w"then990                       <fj>
940 ifx$="z"then760                       <nl>
950 ifx$="m"then200                       <ii>
960 ifx$="h"thengosub2260:goto910         <pj>
970 goto920                               <jk>
980 rem*berechn. anteilige warmwas
serkosten                                 <bm>
990 scnclr:char0,23,0,te$:print:pr
int:printchr$(2);:print"D.) Berech
nung der anteiligen Warmwasserkost
en"                                       <ch>
1000 print:print                          <mn>
1010 print"1. Die Gesamtwarmwasser
kosten aus C 6.) betragen        :
    ";:printusing"#######.># DM";h
z                                         <hf>
1020 print                                <ji>
1030 print"a. Von diesen Kosten en
tfallen 40 % auf Grundkosten-,  :
    ";:printusing"#######.># DM";p
g                                         <cf>
1040 print                                <mb>
1050 print"b. sowie 60 % auf Verbr
auchskosten Warmwasser          :
    ";:printusing"#######.># DM";w
p                                         <co>
1060 print                                <oj>
```

```
1070 input"3. Gesamtwohnflaeche de
s Gebaeudes lt. Abrechnung       :
          ";gw                        <el>
1080 printspc(62);ch$;:printusing"
######.># QM";gw                      <kh>
1090 print                            <cf>
1100 input"4. Gesamtwohnflaeche de
r eigenen Wohnung                :
          ";ww                        <cd>
1110 printspc(62);ch$;:printusing"
######.># QM";ww                      <fg>
1120 print                            <gb>
1130 at=pg/gw*ww                      <bc>
1140 print"5. Diese ergeben anteil
ige Grundkosten Warmwasser       :
     ";ro$;:printusing"#######.># D
M";at                                 <og>
1150 print                            <jo>
1160 input"6. Verbrauchte Einheite
n fuer eigene Wohnung            :
          ";he                        <fj>
1170 printspc(62);ch$;:printusing"
######.># EH";he                      <na>
1180 print                            <nk>
1190 av=wp/vb*he                      <pk>
1200 print"7. Diese ergeben anteil
ige Verbrauchskosten Warmwasser :
     "ro$;:printusing"#######.># DM
";av                                  <ch>
1210 print                            <bg>
1220 gww=at+av-x                      <mc>
1230 print"8. Ihre Gesamtkosten fu
er Warmwasser betragen ( 5 + 7 ):
     ";be$;:printusing"#######.># D
M";gww                                <je>
1240 printba$                         <ea>
1250 gosub3160                        <jm>
1260 getkeyx$                         <mo>
1270 ifx$="w"then1330                 <kb>
1280 ifx$="z"then990                  <lk>
1290 ifx$="m"then200                  <bo>
1300 ifx$="h"thengosub2260:goto125
0                                     <pl>
1310 goto1260                         <ol>
1320 rem*berechn. heizkosten + ant
eil grundkosten                       <nl>
1330 scnclr:char0,23,0,te$:print:p
rint:printchr$(2);:print"E.) Berec
hnung Heizkosten + Anteil Grundkos
ten"                                  <ap>
1340 print:print                      <hj>
1350 print"1. Die Gesamtkosten der
 Heizanlage aus A 6.) betragen  :
     ";:printusing"#######.># DM";g
h                                     <pi>
1360 print                            <ed>
1370 print"2. Abzueglich der Warmw
asserkosten aus C 6.)            :
     ";ro$;:printusing"#######.># D
M";hz                                 <nc>
1380 print                            <gl>
1390 khz=gh-hz                        <gi>
1400 print"3. Somit ergeben sich v
erbleibende Kosten fuer Heizung :
     ";:printusing"#######.># DM";k
hz                                    <po>
1410 print:ghz=khz*40/100             <bk>
1420 print"4. Von diesen Kosten en
tfallen 40 % auf Grundkosten-,   :
     ";:printusing"#######.># DM";g
hz                                    <je>
1430 print:vhz=khz*60/100             <hf>
1440 print"5. sowie 60 % auf den V
erbrauch fuer Heizung            :
     ";:printusing"#######.># DM";v
hz                                    <ap>
1450 print                            <pi>
1460 input"6. Gesamtwohnflaeche de
s Gebaeudes lt. Abrechnung       :
          ";gw                        <jo>
1470 printspc(62);ch$;:printusing"
######.># QM";gw                      <jm>
1480 print                            <de>
1490 input"7. Gesamtwohnflaeche de
r eigenen Wohnung                :
          ";ww                        <pe>
1500 printspc(62);ch$;:printusing"
######.># QM";ww                      <pb>
1510 agh=ghz/gw*ww                    <lf>
1520 print                            <ie>
1530 print"8. Die anteiligen Grund
kosten f. Heizung betragen       :
     ";be$;:printusing"#######.># D
M";agh                                <pj>
1540 printba$                         <bm>
1550 gosub3160                        <fe>
1560 getkeyx$                         <kh>
1570 ifx$="w"then1630                 <le>
1580 ifx$="z"then1330                 <ag>
1590 ifx$="m"then200                  <fk>
1600 ifx$="h"thengosub2260:goto155
0                                     <kl>
1610 goto1560                         <in>
1620 rem*berechn. heizkosten + ant
eil verbrauch                         <hg>
1630 scnclr:char0,23,0,te$:print:p
rint:printchr$(2);:print"F.) Berec
hnung Heizkosten + Anteil Verbrauc
h"                                    <le>
1640 print:print                      <ob>
1650 gh=bk+hn+zk                      <od>
1660 print"1. Die Gesamtkosten der
 Heizanlage aus A 6.) betragen  :
     ";:printusing"#######.># DM";g
h                                     <io>
1670 print                            <lc>
```

```
1680 print"2. Abzueglich der Warmw
asserkosten aus C 6.)          :
    ";ro$;:printusing"#######.># D
M";hz                                <kp>
1690 print                           <nk>
1700 print"3. Somit ergeben sich v
erbleibende Kosten fuer Heizung :
    ";:printusing"#######.># DM";k
hz                                   <fe>
1710 print:ghz=khz*40/100            <en>
1720 print"4. Von diesen Kosten en
tfallen 40 % auf Grundkosten-,   :
    ";:printusing"#######.># DM";g
hz                                   <km>
1730 print                           <ck>
1740 print"5. sowie 60 % auf den V
erbrauch fuer Heizung            :
    ";:printusing"#######.># DM";v
hz                                   <fp>
1750 print                           <fc>
1760 input"6. Gesamtverbrauch der
Einheiten lt. Rechnungssteller   :
     ";geh                           <kb>
1770 printspc(62);ch$;:printusing"
######.># EH";geh                    <gi>
1780 print                           <io>
1790 input"7. Verbrauch lt. Verbra
ucherfassung der Wohnung         :
     ";hew                           <jk>
1800 printspc(62);ch$;:printusing"
######.># EH";hew                    <pm>
1810 vae=vhz/geh*hew                 <bd>
1820 print                           <np>
1830 print"8. Die anteiligen Verbr
auchskosten f. Heizung betragen :
    ";be$;:printusing"#######.># D
M";vae                               <po>
1840 printba$                        <oh>
1850 print                           <bl>
1860 gosub3160                       <ci>
1870 getkeyx$                        <bc>
1880 ifx$="w"then1940                <bb>
1890 ifx$="z"then1630                <jb>
1900 ifx$="m"then200                 <la>
1910 ifx$="h"thengosub2260:goto186
0                                    <po>
1920 goto1870                        <ja>
1930 rem*aufstellung kosten          <io>
1940 scnclr:printu$;tab(8);te$;:pr
int" * AUFSTELLUNG DER ANTEILIGEN
KOSTEN"                              <fc>
1950 print                           <oe>
1960 print"Ihre anteiligen Kosten
setzen sich demnach wie folgt zusa
mmen:":print                         <hb>
1970 printu$;:print"Fuer Warmwasse
r:":print                            <nh>
1980 gww=at+av-x                     <kh>
1990 print"1. Anteilige Grundkoste
n fuer Warmwasser      ( D 5.)  :
    ";:printusing"#######.># DM";a
t                                    <jo>
2000 print"2. Anteilige Verbrauchs
kosten fuer Warmwasser ( D 7.)  :
    ";:printusing"#######.># DM";a
v                                    <pg>
2010 print                           <fm>
2020 print"3. Der Gesamtanteil fue
r Warmwasser betraegt ( 1 + 2)   :
    ";ro$;:printusing"#######.># D
M";gww                               <al>
2030 print                           <ie>
2040 printu$;:print"Fuer Heizung:"
:print                               <hc>
2050 print"4. Anteilige Grundkoste
n fuer Heizung         ( E 8.)  :
    ";:printusing"#######.># DM";a
gh                                   <gd>
2060 print"5. Anteilige Verbrauchs
kosten fuer Heizung    ( F 8.)  :
    ";:printusing"#######.># DM";v
ae                                   <np>
2070 print:gkh=agh+vae               <gj>
2080 print"6. Gesamtanteil fuer He
izung betraegt         ( 4 + 5)  :
    ";ro$;:printusing"#######.># D
M";gkh                               <oe>
2090 print:ggk=gww+gkh               <de>
2100 print"7. Die Gesamtkosten War
mwasser + Heizung betragen somit:
    ";be$;:printusing"#######.># D
M";ggk                               <df>
2110 print                           <cf>
2120 input"8. Abzueglich geleistet
er Vorauszahlungen               :
      ";glz                          <fc>
2130 printspc(62);ch$;:printusing"
######.># DM";glz                    <gl>
2140 su=glz-ggk                      <an>
2150 ifsu>0then2170                  <jn>
2160 ifsu<0then2190                  <aj>
2170 print:print"9. Die Gutschrift
 betraegt abzueglich Ihrer Zahlung
en  :      ";ro$;:printusing"######
#.># DM";su                          <nk>
2180 goto2200                        <ba>
2190 print:print"9. Ihre Nachzahlu
ng betraget abzueglich Ihrer Zahlu
ngen:     ";ro$;:printusing"######
#.># DM";su                          <ik>
2200 gosub 3180                      <ol>
2210 getkeyx$                        <go>
2220 ifx$="m"then200                 <kc>
2230 ifx$="z"then1940                <le>
2240 ifx$="h"thengosub2260:goto220
0                                    <lb>
```

```
2250 goto2210                                  <mp>
2260 char0,2,23,"              Bitte d
en Drucker einschalten und eine Ta
ste druecken !              "                  <hh>
2270 getkeyx$                                  <ck>
2280 char0,2,23,"


                             "                 <kh>
2290 fast:close4:open4,4,7:open3,3
:printchr$(19);                                <ie>
2300 forz=1to25:b$="":k=79:fors=1t
ok:get#3,a$:b$=b$+a$:nexts                     <ph>
2310 ifmid$(b$,k,1)=""thenk=k-1:b$
=left$(b$,k):ifk>0then2310                     <oe>
2320 print#4,b$:nextz:close3:close
4:return                                       <ge>
2330 rem*prg-info                              <gd>
2340 scnclr:printtab(17)u$;te$;:pr
int" - PROGRAMMINFORMATION":print:
print                                          <gn>
2350 print"Fuer viele ist die jaeh
rliche vorgelegte Heizkostenabrech
nung ein Buch mit sie- ben Siegeln
 und wird von den meisten so hinge
nommen und ohne Kommen";                       <oh>
2360 print"tar bezahlt."                       <mo>
2370 print                                     <cp>
2380 print"Das dem nicht so sein m
uss, zeigt dieses Heizkostenabrech
nungsprogramm, welche  die Angaben
 des Rechnungsstellers auf seine R
ichtigkeit ueberprueft."                       <ci>
2390 print                                     <fh>
2400 print"Die rechtliche Grundlag
e zur Heizkostenabrechnung ist die
 Verordnung ueber Heiz-kostenabrec
hnungen vom 05.04.1984. Bekanntgeg
eben im Bundesgesetz";                         <nh>
2410 print"tzblatt Teil I Seite 59
2 vom 12.04.1984"                              <lb>
2420 print                                     <jd>
2430 print"Eine Broschuere zu dem
Thema Heizkostenabrechnung kann be
im Bundeswirtschafts-  ministerium
 kostenlos angefordert werden."               <cj>
2440 print                                     <lm>
2450 print"Der Titel der Broschuer
e lautet:":print                               <be>
2460 printro$;be$;:printtab(10)"Ve
rbrauchsabhaengige Abrechnung - He
iz- u. Warmwasserkosten":printrf$;
ba$                                            <fa>
2470 print"Dort koennen alle weite
ren wichtigen Informationen entnom
men werden."                                   <jg>
2480 print:printtab(17)"Bitte drue
cken Sie die Taste "+ro$+"M"+rf$+"
 enue";:print"  "+ro$+"W"+rf$+" ei
ter"                                           <lc>
2490 getkeyx$                                  <pj>
2500 ifx$="m"then200                           <cm>
2510 ifx$="w"then2530                          <hp>
2520 goto2490                                  <mj>
2530 scnclr:printtab(17)u$;te$;:pr
int" - PROGRAMMINFORMATION":print:
print                                          <kd>
2540 print"Hier erhalten Sie einen
 kurzen Ueberblick ueber die wicht
igsten Daten der Ver-  ordnung.               <pd>
2550 print                                     <ji>
2560 print"Der Paragraph"+be$+" 9
"+ba$+"regelt die Art und Weise de
r abzurechnenden Kosten fuer Heiza
n- lage und Warmwasser nach vorgeg
ebener festen Formel."                         <bf>
2570 print                                     <mb>
2580 printtab(26)"B="+u$+" 2,5 x V
 + (tw - 10)"                                  <nb>
2590 printtab(36)"Hu"                          <an>
2600 print                                     <pn>
2610 print"B    = Brennstoffverbrau
ch (in cbm oder kg)":print                     <mi>
2620 print"2,5 = Faktor ist eine e
rmittelte Konstante aus mittleren
Jahresnutzungsgrad":print                      <jk>
2630 print"V    = Menge des verbrau
chten Warmwasser in cbm"                       <dd>
2640 print"Hu   = Heizwert des Bren
nstoffes nach Angaben des Versorgu
ngsunternehmens"                               <dk>
2650 print"tw   = gemessene mittler
e Temperatur des Warmwassers. (min
d. 60 Grad Celsius)"                           <fb>
2660 print"       (Da die uebliche
Zuflusstemperatur des Kaltwassers
ca.10 Grad Celsius be-         trae
gt, ist diese von 60 Grad abzuzieh
en)                                            <ae>
2670 print:printtab(14)"Bitte drue
cken Sie die Taste "+ro$+"M"+rf$+"
 enue";:print"  "+ro$+"W"+rf$+" ei
ter";:print"  "ro$+"Z"+rf$+" ureuc
k"                                             <ke>
2680 getkeyx$                                  <on>
2690 ifx$="m"then200                           <cc>
2700 ifx$="z"then2340                          <cf>
2710 ifx$="w"then2730                          <fh>
2720 goto2680                                  <pe>
2730 scnclr:printtab(17)u$;te$;:pr
int" - PROGRAMMINFORMATION":print:
print                                          <jm>
2740 print                                     <bg>
2750 print"Das vorliegende Program
m beruecksichtigt alle Formen der
Verordnung und ist     fuer den no
rmalen Abrechnungsmodus - Mietwohn
```

```
ungen eingerichtet."                   <dk>
2760 print                             <do>
2770 print"Um das Programm zu nutz
en, sind nur die Daten aus der Hei
zkostenabrechnung ein- zugeben; al
les weitere regelt bzw. berechnet
(ueberprueft) das";                    <oe>
2780 print" Programm."                 <jp>
2790 print                             <hk>
2800 print"Die Richtigkeit der Ang
aben Ihres Vermieters, kann das Pr
ogramm Ihnen natuer-   lich nicht
abnehmen."                             <mm>
2810 print                             <kb>
2820 print"Sehen Sie daher unbedin
gt in die Abrechnungsunterlagen (R
echtsanspruch) Ihres   Vermieters
ein."                                  <od>
2830 print                             <ml>
2840 printtab(8)"Geben Sie nun Ihr
e aus der Abrechnung ersichtlichen
 Daten ein."                           <nh>
2850 print                             <pd>
2860 print"P.S. - Erzielen Sie kei
ne Uebereinstimmung, passe ich Ihn
en das Programm gern   nach Zusend
ung einer Kopie Ihrer Abrechnungsu
nterlagen an."                         <gg>
2870 print                             <bl>
2880 print:printtab(17)"Bitte drue
cken Sie die Taste "+ro$+"M"+rf$+"
 enue";:print"  "+ro$+"Z"+rf$+" ur
ueck"                                  <ma>
2890 getkeyx$                          <bc>
2900 ifx$="m"then200                   <jf>
2910 ifx$="z"then2530                  <jf>
2920 goto2880                          <ab>
2930 rem*programmende                  <im>
2940 scnclr:char0,24,12,"Sind Sie
sicher Ä"+be$+"J/N"+ba$+"Ü"            <ok>
2950 getkeyx$                          <no>
2960 ifx$="j"then2990                  <fk>
2970 ifx$="n"then100                   <af>
2980 goto2950                          <jh>
2990 scnclr:char0,18,12,"Aufwieder
sehen bis zum naechsten Mal !"         <nn>
3000 sys65341                          <bh>
3010 scnclr:printzf$;:fora=1to78:p
rintzv$;:next:char1,79,0,zd$           <bf>
3020 printyo$;:fora=1to78:print" "
;:next:char1,79,1,yo$                  <db>
3030 printyo$;:printtab(8)te$;tab(
42)"ÄCÜ by G. Kl.-Uebbing > 1987 <
":char1,79,2,yo$                       <ni>
3040 printyo$;:fora=1to78:print" "
;:next:char1,79,3,yo$                  <if>
3050 printzc$;:fora=1to78:printzv$
;:next:char1,79,4,zs$                  <mn>
3060 printchr$(27)+"m"                 <bi>
3070 print                             <kl>
3080 char1,25,8,rn$+" 1 "+rf$+" _
 Eingabe Daten aus Abrechnung"         <bg>
3090 char1,25,10,rn$+" 2 "+rf$+" _
 Programminformation"                  <pl>
3100 char1,25,14,rn$+" 3 "+rf$+" _
 Programm beenden"                     <bi>
3110 fora=1to4:print:next              <jc>
3120 printzf$;:fora=1to78:printzv$
;:next:char1,79,18,zd$:print           <ji>
3130 printyo$;:printtab(32)"> "aa$
"Bitte Eingabe":printtab(48)c2$"<"
;:printtab(79)yo$                      <ie>
3140 printzc$;:fora=1to78:printzv$
;:next:char1,79,20,zs$:print           <dd>
3150 print:return                      <dh>
3160 char0,2,23,"Bitte Taste druec
ken >>> "+ro$+" W "+rf$+" eiter
"+ro$+" Z "+rf$+" urueck   "+ro$+"
 M "+rf$+" enue   "+ro$+" H "+rf$+
" ardcopy":print                       <el>
3170 return                            <gk>
3180 char0,7,23,"Bitte Taste druec
ken >>> "+ro$+" Z "+rf$+" urueck
 "+ro$+" M "+rf$+" enue   "+ro$+"
H "+rf$+" ardcopy":print               <em>
3190 return                            <jd>
60000 rem nachspann ============= <pj>
60010 rem *farbcodes/steuercodes * <kg>
60020 aa$=chr$(015):rn$=chr$(018)  <dc>
60030 c2$=chr$(145):rf$=chr$(146)  <pe>
60040 rem ***zeichensatz/graphik * <lm>
60050 zc$=chr$(173):zd$=chr$(174)  <ip>
60060 zf$=chr$(176):zs$=chr$(189)  <pk>
60070 zv$=chr$(192):yo$=chr$(221)  <jb>
60080 return                       <md>
60090 rem ende                     <lk>
```

OIL-COMPANY

In diesem Spiel schlüpft man in die Rolle eines Managers, der mit einer Million Dollar anfängt, um einen Ölkonzern aufzubauen. Dabei muß man eine Nase für günstige Gelegenheiten haben. Und natürlich spielt dabei der Zufall auch eine nicht unwichtige Rolle.

BOHRUNGEN

Bohrrechte zu erwerben und dann nach Öl zu bohren sind natürlich die ersten Schritte, um Milliardär zu werden. Sie haben dazu pro Runde drei mal die Gelegenheit.

BESTANDSAUFNAHME

Anhand eines Balkendiagrammes können Sie hier die Ölmengen in Ihren Lagern mit einem Blick erfassen.

TRANSPORT/VERKAUF

Hier können Sie Ihr „Schwarzes Gold" per Ölfrachter nach Rotterdam, Kawasaki oder Philadelphia transportieren, um es zu verkaufen. Es empfiehlt sich, den Hafen zu nehmen, wo der Marktpreis am höchsten ist. Wenn Sie viel Öl verkaufen, werden sich Ihre Mitspieler nicht gerade freuen, denn je mehr Sie umsetzen, desto stärker sinkt der Ölpreis.

KREDIT

Es ist ratsam, zu Beginn eines Spiels erst einmal einen Kredit aufzunehmen; zu einem Zinssatz von 15 %. Wichtig!! : Verlassen Sie nicht die Runde, wenn Sie mit Ihrem rechts oben angezeigten Kapital im Soll stehen. Sie werden sonst mit äußerst massiven Überzugszinsen dafür bestraft!

PUNKTE

Wenn Sie diesen Menüpunkt anwählen, wird Ihnen gezeigt, wie weit Sie schon auf Ihrem Weg zum Großkonzern sind.

NÄCHSTER SPIELER

Um die Runde zu beenden, wählen Sie diesen Punkt an. Gleichzeitig werden Ihnen die laufenden Kosten für Ihre Ölfelder abgezogen.

ENDE

Hier beenden Sie Ihr Spiel. Nicht verkaufte Ölmengen sowie Ölfelder werden Ihnen vergütet und Schulden abgezogen.

Vorschlag: Der Gerechtigkeit wegen ist es ratsam, zu Beginn zu vereinbaren, bis zu welchem Jahr man spielt.

(Ingo Rösler)

```
10 rem oil-company===========1
28                               <pl>
20 rem (p) 07/87 commodore wel
t=                               <gh>
30 rem ======================
==                               <kc>
40 rem (c) 07/87 by
==                               <gf>
50 rem            ingo roesler
==                               <ie>
60 rem
==                               <ef>
70 rem version 7.0 40z/din-t.
==                               <ib>
80 rem 128 pc+kass./floppy
==                               <bl>
90 rem ======================
==                               <no>
95 gosub 60000                   <oc>
100 poke0,peek(0) or 64          <le>
110 poke1,peek(1) and 191:poke
808,100                          <bo>
120 graphic1,1:fast              <eb>
130 print chr$(27)+"e"           <mp>
140 color0,2:color4,6:color5,6
:color1,6                        <ce>
150 print chr$(14)chr$(11)       <cg>
160 da=1980                      <kj>
170 rem *** titelbild ******** <ke>
180 draw1,157,20 to 163,20 to
163,27 to 170,27 to 175,60 to
180,60 to 180,65 to 176,65 to
180,100 to 187,100               <nl>
190 draw1,187,100 to 187,105 t
o 173,105 to 173,100 to 177,10
0 to 173,65 to 147,65 to 143,1
00 to 147,100                    <ad>
200 draw1,147,100 to 147,105 t
o 133,105 to 133,100 to 140,10
0 to 144,65 to 140,65 to 140,6
5 to 140,60 to 145,60            <fp>
210 draw1,145,60 to 150,27 to
157,27 to 157,20                 <le>
220 draw1,153,30 to 167,30 to
172,60 to 148,60 to 153,30,153
,30 to 172,60,148,60 to 167,30 <ng>
230 draw1,147,65 to 175,80 to
145,80 to 173,65                 <ni>
240 draw1,175,80 to 144,95 to
176,95  to 145,80                <oi>
250 draw1,158,30 to 158,60,162
,30 to 162,60                    <ga>
260 draw1,158,65 to 158,107,16
2,65 to 162,107                  <np>
270 draw1,151,42 to 169,42,158
,107 to 162,107                  <np>
280 slow                         <dn>
290 char1,13,17,"oil company"  <md>
300 char1,9,19,"by ingo r"+zr$
+"sler (c)1986"                  <mn>
310 sleep2:getkey a$:graphic0  <pd>
320 scnclr:print:print:print:p
rint chr$(14)                    <pc>
330 print" Wieviele Spieler ne
hmen teil (1-6) ?"               <kd>
340 trap                         <cm>
350 getkey z$:if asc(z$)<49 or
 asc(z$)>54 then 350             <ip>
360 z=val(z$)                    <aj>
370 print:open1,0                <hc>
380 for i=1 to z                 <ci>
390 print" "chr$(18)"Name"chr$
(146)": ";:input#1,na$(i):prin
t                                <hb>
400 v=len(na$(i))                <pk>
410 if v<3 or v>8 then print c
hr$(145)chr$(145):goto390        <fe>
420 ka(i)=1000000                <mo>
430 next i:s=1:close1            <pf>
440 mr=320000:mk=390000:mp=270
000                              <jb>
450 rem *** menue ************ <ec>
460 scnclr:print:print           <fl>
470 w=1                          <bb>
480 print" "na$(s)tab(10)"U.S.
-Dollar:";:print using "#,###,
###,###";ka(s):print             <jm>
490 printtab(10)"Jahr:"da        <fj>
500 print" ______________________
________________________"        <om>
510 char1,3, 9," $$$  $$ $$"     <ea>
520 char1,3,10,"$$   $ $$ $$"    <hd>
530 char1,3,11,"$$   $ $$ $$"    <ho>
540 char1,3,12,"$$   $ $$ $$"    <ij>
550 char1,3,13,"$$   $ $$ $$"    <je>
560 char1,3,14,"$$   $ $$ $$"    <jp>
570 char1,3,15," $$$  $$ $$$$$
"                                <bg>
580 char1,1,19,"_________________
__"                              <kk>
590 gosub 3470:char1,20,8,"Boh
rung             ",1             <ek>
600 poke208,0:getkey a$:gosub
3470                             <lf>
610 if a$=chr$(017) then w=w+1 <ji>
620 if a$=chr$(145) then w=w-1 <kf>
630 if a$=chr$(013) then 730   <ah>
640 if w>7 then w=1              <cn>
650 if w<1 then w=7              <dj>
660 if w=1 then char1,20,8,"Bo
hrung            ",1:goto600     <ma>
670 if w=2 then char1,20,10,"B
estandsaufnahme ",1:goto600      <hi>
680 if w=3 then char1,20,12,"T
ransport/Verkauf",1:goto600      <hm>
```

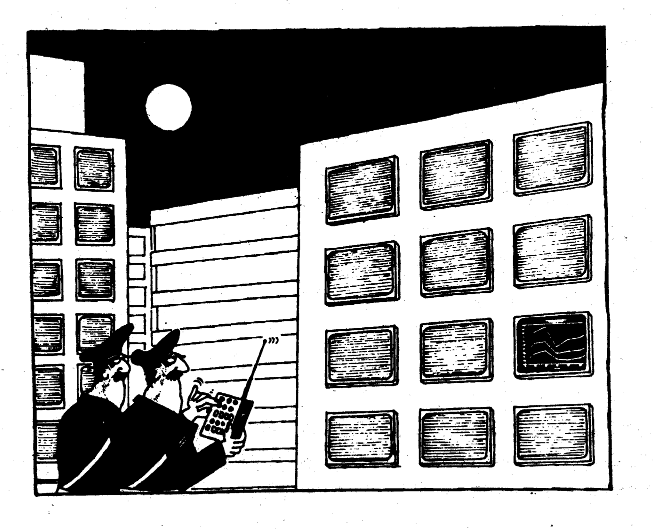
```
690 if w=4 then char1,20,14,"K
redit              ",1:goto600     <lm>
700 if w=5 then char1,20,16,"P
unkte              ",1:goto600     <nn>
710 if w=6 then char1,20,18,"N
"+zq$+"chster Spieler ",1:goto
600                                <cf>
720 if w=7 then char1,20,20,"E
nde                ",1:goto600     <dd>
730 on w goto 740,2090,1240,19
00,3100,2900,2560                  <ki>
740 rem *** bohrung **********     <km>
750 yb=yb+1:if yb>3 then 460       <di>
760 scnclr:print:restore           <hl>
770 o=int(rnd(0)*10)+1:if o>6
then 840                           <go>
780 for x=1 to 6                   <kf>
790 read bo$(x),p(x)               <cd>
800 next x                         <pn>
810 bo$=bo$(o)                     <oj>
820 p=p(o)                         <cj>
830 data"Iran",890000,"Lybien"
,1490000,"Nigeria",1590000,"Ve
nezuela",1230000,"Nigeria",800
000,"Alaska",1100000               <ef>
840 if o>6 then print" Bohrung
 mi"zt$"gl"zs$"ckt !":ka(s)=ka
(s)-867000:getkey a$:goto460       <do>
850 print" "na$(s)tab(10)"U.S.
-Dollar:";:print using "#,###,
###,###";ka(s):print               <ba>
860 printtab(10)"Jahr:"da          <mn>
870 print" ______________________
_________________________":print:pri
nt                                 <ag>
880 print" Angebot in ";bo$        <fh>
890 print" zum Preis von $";:p
rint using "#,###,###,###";p       <ck>
900 print                          <ca>
910 print" Nehmen Sie das Ange
bot an ? (J/N)                     <cg>
920 getkey a$                      <jk>
930 if a$="j" then begin else
goto650                            <il>
940 xb=xb+1                        <ha>
950 if o=1 then sa(s)=sa(s)+1      <bk>
960 if o=2 then ir(s)=ir(s)+1      <dd>
970 if o=3 then li(s)=li(s)+1      <dc>
980 if o=4 then ni(s)=ni(s)+1      <eb>
990 if o=5 then ve(s)=ve(s)+1      <fe>
1000 if o=6 then al(s)=al(s)+1     <ed>
1010 ka(s)=ka(s)-p                 <ac>
1020 print                         <ji>
1030 print" Ihr jetziges Kapit
al $";:print using "#,###,###,
###";ka(s)                         <pd>
1040 getkey a$                     <bd>
1050 bend                          <df>
1060 goto460                       <el>
1070 rem *** ereignis ********     <of>
1080 ye=ye+1:if ye>3 then 460      <in>
1090 scnclr:print:print chr$(1
4)                                 <fb>
1100 print" "na$(s)tab(10)"U.S
.-Dollar:";:print using "#,###
,###,###";ka(s):print              <al>
1110 printtab(10)"Jahr:"da         <mi>
1120 print" ____________________
_________________________":print:pr
int                                <ab>
1130 en=int(rnd(0)*8)+1            <le>
1140 restore 3560                  <nc>
1150 for i=1 to 8                  <al>
1160 read en$(i),oe(i),k(i),of
(i)                                <ck>
1170 next i                        <gb>
1180 print en$(en)                 <ne>
1190 sa(s)=sa(s)+of(en)            <op>
1200 os(s)=os(s)+oe(en)            <be>
1210 ka(s)=ka(s)+k(en):getkey
x$                                 <pj>
1220 if s=z then s=1:da=da+1:g
oto460                             <ln>
1230 s=s+1:goto460                 <gc>
1240 rem *** transport *******     <nm>
1250 rem                           <hf>
1260 scnclr:print:print            <ho>
1270 print" Verkauf":print" --
-----":print:print                 <gd>
1280 print" Land
  "yn$"lmenge in 1000 t"           <pe>
1290 print" ------------------
--------------------"              <jk>
1300 print                         <lc>
1310 print" [S]audi-Arabien...
.";os(s)                           <ig>
1320 print" [I]ran............
.";oi(s)                           <dj>
1330 print" [L]ibyen..........
.";ol(s)                           <id>
1340 print" [N]igeria.........
.";og(s)                           <jg>
1350 print" [V]enezuela.......
.";ov(s)                           <ad>
1360 print" [A]laska..........
.";oa(s)                           <ia>
1370 print" [Q]uit":print          <ik>
1380 print" Aus welchem Land e
xportieren ?"                      <oh>
1390 getkey a$:if a$=chr$(13)
then 1390                          <do>
1400 if a$<>"s" and a$<>"i" an
d a$<>"l" and a$<>"n" and a$<>
"v" and a$<>"a" and a$<>"q" th
en 1390                            <ap>
1410 if a$="q" then 460            <pp>
```

```
1420 print" Ein Schiff fa"zt$"
t 100.000 t":open1,0:print" An
zahl der Schiffe: ";:input#1,s
f$:close1                          <ph>
1430 sf=val(sf$):if sf<1 or sf
>100 or sf<>int(sf) then print
 chr$(145)chr$(145):goto1420       <aa>
1440 if sf$="" then print chr$
(145)chr$(145):goto1420            <bb>
1450 if a$="s" and os(s)<100*s
f then print" Nicht m"+zr$+"gl
ich !":goto1390                    <kn>
1460 if a$="i" and oi(s)<100*s
f then print" Nicht m"+zr$+"gl
ich !":goto1390                    <kd>
1470 if a$="l" and ol(s)<100*s
f then print" Nicht m"+zr$+"gl
ich !":goto1390                    <ld>
1480 if a$="n" and og(s)<100*s
f then print" Nicht m"+zr$+"gl
ich !":goto1390                    <lk>
1490 if a$="v" and ov(s)<100*s
f then print" Nicht m"+zr$+"gl
ich !":goto1390                    <nl>
1500 if a$="a" and oa(s)<100*s
f then print" Nicht m"+zr$+"gl
ich !":goto1390                    <ll>
1510 print:print                   <fh>
1520 print" In welchen Hafen s
oll es verschifft      werden ?
"                                  <jl>
1530 print                         <ji>
1540 print" Zielhafen          Ko
sten   Marktpr./Sch."              <nl>
1550 print" -------------------
--------------------"              <jp>
1560 print                         <lh>
1570 print" [R]otterdam       $
49500  $"mr                        <ni>
1580 print" [K]awasaki        $
58250  $"mk                        <hh>
1590 print" [P]hiladelphia    $
42600  $"mp                        <jh>
1600 print" [Q]uit"                <jo>
1610 getkey h$                     <fg>
1620 if h$="r" then ka(s)=ka(s
)-49500*sf:tr=1:goto1670           <ji>
1630 if h$="q" then goto460        <gm>
1640 if h$="k" then ka(s)=ka(s
)-58250*sf:tk=1:goto1670           <ka>
1650 if h$="p" then ka(s)=ka(s
)-58250*sf:tp=1:goto1670           <le>
1660 goto 1610                     <nd>
1670 if a$="s" then os(s)=os(s
)-100*sf                           <cl>
1680 if a$="i" then oi(s)=oi(s
)-100*sf                           <bh>
1690 if a$="l" then ol(s)=ol(s
)-100*sf                           <ck>
1700 if a$="n" then og(s)=og(s
)-100*sf                           <cm>
1710 if a$="v" then ov(s)=ov(s
)-100*sf                           <fm>
1720 if a$="a" then oa(s)=oa(s
)-100*sf                           <ch>
1730 rem *** verkauf *********     <gd>
1740 scnclr:print:print:print:     <gn>
1750 printsf;" Schiff(e) ist (
sind) im Zielhafen       angek
ommen"                             <fd>
1760 print" Sie verkaufen die
Ladung zum Preis von  $";          <cj>
1770 if tr=1 then pr=mr            <jo>
1780 if tk=1 then pr=mk            <jk>
1790 if tp=1 then pr=mp            <ko>
1800 if tr=0 and tk=0 and tp=0
 then pr=0                         <lc>
1810 pr=pr*sf                      <fe>
1820 printpr                       <fo>
1830 print                         <mg>
1840 print" Ihr jetziges Kapit
al: $"ka(s)+pr                     <dm>
1850 ka(s)=ka(s)+pr:tr=0:tk=0:
tp=0                               <nm>
1860 if h$="r" then mr=mr-sf*4
0000                               <af>
1870 if h$="k" then mk=mk-sf*5
5000                               <aa>
1880 if h$="p" then mp=mp-sf*7
0000                               <bg>
1890 getkey a$:goto460             <mf>
1900 rem *** kredit **********     <oh>
1910 scnclr:print:print chr$(1
4)                                 <ii>
1920 print" "na$(s)tab(10)"U.S
.-Dollar:";:print using "#,###
,###,###";ka(s):print              <ec>
1930 printtab(10)"Jahr:"da         <pp>
1940 print" ___________________
_____________________":print:pr
int                                <di>
1950 print" Wollen Sie einen K
redit ?"s2$"(J/N)"                 <ck>
1960 getkey a$:if a$="j" then
1980                               <jb>
1970 goto460                       <nm>
1980 print                         <fm>
1990 print" In welcher H"zr$"h
e ";:inputkd                       <ln>
2000 if kd<0 then printchr$(14
5)chr$(145):goto1990               <ao>
2010 kd=int(kd/1000)*1000          <jc>
2020 print" Sie haben jetzt $"
ka(s)+kd                           <kf>
2030 print:kr(s)=kr(s)+kd          <df>
2040 print" Am Ende des Spiele
```

```
s wird Ihnen insges."                    <bg>
2050 print:print" $";int(kr(s)
*1.15):print                             <ke>
2060 print" von Ihrem Kapital
abgezogen."                              <bp>
2070 ka(s)=ka(s)+kd:getkey a$            <kc>
2080 goto460                             <el>
2090 rem * bestandsaufnahme **           <ni>
2100 graphic1,1                          <ki>
2110 char1,13,3,"lagerbest"+zq
$+"nde"                                  <in>
2120 char1,13,4,"-------------
"                                        <ak>
2130 if os(s)>3000 then os(s)=
3000                                     <ab>
2140 if oi(s)>3000 then oi(s)=
3000                                     <ph>
2150 if ol(s)>3000 then ol(s)=
3000                                     <ah>
2160 if og(s)>3000 then og(s)=
3000                                     <ah>
2170 if ov(s)>3000 then ov(s)=
3000                                     <cp>
2180 if oa(s)>3000 then oa(s)=
3000                                     <ap>
2190 for i=30 to 40                      <hn>
2200 draw1,i,160to i,160-int(o
s(s)/20)                                 <ke>
2210 next i                              <hf>
2220 for i=62 to 72                      <kf>
2230 draw1,i,160to i,160-int(o
i(s)/20)                                 <li>
2240 next i                              <jd>
2250 for i=94 to 104                     <pe>
2260 draw1,i,160to i,160-int(o
l(s)/20)                                 <nj>
2270 next i                              <lb>
2280 for i=126 to 136                    <ed>
2290 draw1,i,160to i,160-int(o
g(s)/20)                                 <pc>
2300 next i                              <mp>
2310 for i=158 to 168                    <gm>
2320 draw1,i,160to i,160-int(o
v(s)/20)                                 <ca>
2330 next i                              <oo>
2340 for i=190 to 200                    <hj>
2350 draw1,i,160to i,160-int(o
a(s)/20)                                 <cj>
2360 next i                              <am>
2370 char1,4,22,"sau"                    <kf>
2380 char1,8,22,"ira"                    <kg>
2390 char1,12,22,"lib"                   <ng>
2400 char1,16,22,"nig"                   <ol>
2410 char1,20,22,"ven"                   <pl>
2420 char1,24,22,"ala"                   <oo>
2430 getkey a$                           <ig>
2440 graphic0:scnclr:print:pri
nt:print:print chr$(14)                  <ik>
2450 print" Besitz:"                     <ad>
2460 print" --------":print              <ja>
2470 print" Land               "yn
$"lfld.  Menge in 1000 t"                <cc>
2480 print" ------------------
-------------------"                     <ee>
2490 print" Saudi-Arabien ";sa
(s)tab(23)os(s)                          <bp>
2500 print" Iran          ";ir
(s)tab(23)oi(s)                          <dl>
2510 print" Libyen        ";li
(s)tab(23)ol(s)                          <nl>
2520 print" Nigeria       ";ni
(s)tab(23)og(s)                          <bo>
2530 print" Venezuela     ";ve
(s)tab(23)ov(s)                          <ol>
2540 print" Alaska        ";al
(s)tab(23)oa(s)                          <na>
2550 getkey a$:goto460                   <fl>
2560 rem *** ende ************            <cl>
2570 rem *** abrechnung ******           <pk>
2580 rem                                 <kn>
2590 scnclr:print:print" Wirkl
ich (J/N) ":getkey j$                    <db>
2600 if j$<>"j" then 460                 <fo>
2610 scnclr:print:print:print:
print                                    <ha>
2620 for s=1 to z                        <pl>
2630 of=(sa(s)+ir(s)+li(s)+ni(
s)+ve(s)+al(s))*500000                   <gg>
2640 om=(os(s)+oi(s)+ol(s)+og(
s)+ov(s)+oa(s))*25000                    <gg>
2650 if kr(s)>0 then ka(s)=ka(
s)-(kr(s)*1.15)                          <hi>
2660 ka(s)=of+om+ka(s)                   <af>
2670 ka(s)=ka(s)+pu(s)*1500              <op>
2680 next s                              <fh>
2690 print" Abrechnung:"                 <aa>
2700 print" -----------"                 <gc>
2710 print                               <dj>
2720 c=1:if z=1 then 2840                <eh>
2730 for j=1 to z-1                      <dm>
2740 if ka(j)>=ka(j+1) then 28
20                                       <ei>
2750 kr=ka(j)                            <do>
2760 ka(j)=ka(j+1)                       <kn>
2770 ka(j+1)=kr                          <cn>
2780 z$=na$(j)                           <ge>
2790 na$(j)=na$(j+1)                     <bj>
2800 na$(j+1)=z$                         <fd>
2810 c=0                                 <cj>
2820 next j                              <nl>
2830 if c=0 then 2720                    <dl>
2840 for i=1 to z                        <mo>
2850 printi;chr$(157)".";na$(i
)                                        <pj>
2860 print"   $";:print using
"#,###,###,###";ka(i)                    <mi>
```

```
2870 print" -----------------
---"                                    <mp>
2880 next i                             <bg>
2890 poke808,110:end                    <ae>
2900 rem *** naechster spieler
 ***                                    <jl>
2910 yb=0:ye=0:xb=0                     <fg>
2920 if ka(s)<0 then su=0-ka(s
):ka(s)=ka(s)-(su*3.2)                  <em>
2930 if sa(s)>0 then os(s)=os(
s)+100*sa(s):ka(s)=ka(s)-2500*
os(s)                                   <nf>
2940 if ir(s)>0 then oi(s)=oi(
s)+ 80*ir(s):ka(s)=ka(s)-2500*
oi(s)                                   <kg>
2950 if li(s)>0 then ol(s)=ol(
s)+120*li(s):ka(s)=ka(s)-1500*
ol(s)                                   <nh>
2960 if ni(s)>0 then og(s)=og(
s)+110*ni(s):ka(s)=ka(s)-2500*
og(s)                                   <ng>
2970 if ve(s)>0 then ov(s)=ov(
s)+160*ve(s):ka(s)=ka(s)-1500*
ov(s)                                   <bj>
2980 if al(s)>0 then oa(s)=oa(
s)+70*al(s):ka(s)=ka(s)-1500*o
a(s)                                    <ji>
2990 pb=0                               <cn>
3000 mr=mr+(int(rnd(0)*600000)
-250000)                                <je>
3010 if mr<50000 then mr=50000          <im>
3020 if mr>1200000 then mr=120
0000                                    <fa>
3030 mk=mk+(int(rnd(0)*500000)
-210000)                                <jp>
3040 if mk<50000 then mk=50000          <jm>
3050 if mk>1000000 then mk=100
0000                                    <fm>
3060 mp=mp+(int(rnd(0)*400000)
-180000)                                <mm>
3070 if mp<50000 then mp=50000          <me>
3080 if mp>800000 then mp=8000
00                                      <dd>
3090 goto1070                           <gp>
3100 rem *** punkte **********          <la>
3110 rem *********************          <dd>
3120 pb=pb+1:if pb>1 then 3350          <ji>
3130 scnclr:print:print                 <ne>
3140 if ka(s)>9900000000 then
pu(s)=pu(s)+100:goto3260                <gc>
3150 if ka(s)>7000000000 then
pu(s)pu(s)+94:goto3260                  <il>
3160 if ka(s)>5000000000 then
pu(s)=pu(s)+80:goto3260                 <ea>
3170 if ka(s)>2000000000 then
pu(s)=pu(s)+60:goto3260                 <ef>
3180 if ka(s)> 800000000 then
pu(s)=pu(s)+50:goto3260                 <ce>
3190 if ka(s)> 400000000 then
pu(s)=pu(s)+45:goto3260                 <co>
3200 if ka(s)> 200000000 then
pu(s)=pu(s)+31:goto3260                 <db>
3210 if ka(s)> 100000000 then
pu(s)=pu(s)+25:goto3260                 <dn>
3220 if ka(s)>  10000000 then
pu(s)=pu(s)+20:goto3260                 <bc>
3230 if ka(s)>   5000000 then
pu(s)=pu(s)+17:goto3260                 <pg>
3240 if ka(s)>   1000000 then
pu(s)=pu(s)+10:goto3260                 <pf>
3250 om=os(s)+oi(s)+ol(s)+og(s
)+ov(s)+oa(s)                           <ng>
3260 oe=os(s)+oi(s)+ol(s)+og(s
)+ov(s)+oa(s)                           <ni>
3270 pu(s)=pu(s)+(oe/100)               <od>
3280 of=sa(s)+ir(s)+li(s)+ni(s
)+ve(s)+al(s)                           <na>
3290 pu(s)=pu(s)+(of*15):if pu
(s)>3200 then pu=3200                   <bd>
3300 pu(s)=pu(s)-int(kr/250000
)                                       <fg>
3310 if yb=0 then 3350                  <he>
3320 if xb=0 then pu(s)=pu(s)-
47                                      <am>
3330 if xb=1 then pu(s)=pu(s)-
34                                      <be>
3340 if xb=2 then pu(s)=pu(s)-
21                                      <bl>
3350 graphic1,1                         <ip>
3360 fast:for x=50 to 100               <kp>
3370 draw1,x,150 to x,150-int(
pu(s)/35)                               <ga>
3380 next x                             <bl>
3390 draw1,20,150 to 130,150:s
low                                     <aj>
3400 pu$=str$(int(pu(s)))               <de>
3410 char1,17,2,"punkte"                <kj>
3420 char1,17,3,"------"                <ol>
3430 char1,2,4,"punktzahl:"             <kn>
3440 char1,2,6,(pu$)                    <je>
3450 getkey a$                          <ig>
3460 graphic0:goto460                   <pm>
3470 rem *** menuepunkte                <pj>
3480 char1,20,8,"Bohrung
     "                                  <kn>
3490 char1,20,10,"Bestandsaufn
ahme "                                  <ge>
3500 char1,20,12,"Transport/Ve
rkauf"                                  <gh>
3510 char1,20,14,"Kredit
     "                                  <kg>
3520 char1,20,16,"Punkte
     "                                  <mg>
3530 char1,20,18,"N"+zq$+"chst
er Spieler "                            <an>
3540 char1,20,20,"Ende
```

```
   "                                  <bk>
3550 return                           <hj>
3560 data" Sie kaufen 100.000
t "yn$"l in Saudi-        Arabi
en. Preis $134.000.",100,-1340
00,0                                  <he>
3570 data" "yn$"l ist ausgelau
fen ! Kosten $134.957",0,-1349
57,0                                  <jd>
3580 data" Sie erben ein "yn$"
lfeld in Saudi-Arabien !",0,0,
1                                     <fb>
3590 data" Sie verlieren einen
 Prozess wegen        Steuerhin
terziehung. Sie m"zs$"ssen
     $850.000 Kaution zahlen."
,0,-850000,0                          <db>
3600 data" Sie machen eine Feh
linvestition in       eine Comp
uterfirma.
 Verlust: $1.250.000",0,-12500
00,0                                  <ic>
3610 data" Sie gewinnen einen
Prozess und be-      kommen $1
.500.000",0,1500000,0                 <ec>
3620 data" Einer Ihrer Bohrt"z
s$"rme explodiert.        Zahle
n Sie $600.000 um den Schaden
     zu ersetzen.",0,-600000,0        <jk>
3630 data" Sie legen $3.000.00
0 in Aktien an und    gewinnen
$5.300.000.",0,5300000,0              <ii>
60000 rem nachspann ==========        <nf>
60010 rem zeichensatz/graphik         <ga>
60020 s2$=chr$(160):zq$=chr$(1
87)                                   <nl>
60030 zr$=chr$(188):zs$=chr$(1
89)                                   <bk>
60040 zt$=chr$(190):yn$=chr$(2
20)                                   <al>
60050 return                          <ak>
```



MUSIK-GENIE

Der Name spricht für sich. Dies ist ein Programm, das dem Basic-Programmierer mit geringsten Aufwand gestattet, eigene Musikwerke zu erstellen und – wie bequem – diese auch sofort in Form eines separaten Datenfiles auf Diskette abzuspeichern. Zum Erzeugen und zur Eingabe dieser Musiknoten halten Sie sich am besten an das Handbuch zum C 128, Anhang F. Hier sind nämlich die einzugebenden Parameter aufgeführt. Geben Sie diese einfach ein. In diesen Modus kommen Sie durch das Hauptmenue-Punkt 1 = Neue Musikdatei einrichten.

Es erscheint nun ein Untermenue, in dem Sie ebenfalls wieder Punkt 1 (= Noten eingeben. Hier sollten Sie sich eben diese aus dem Handbuch abtippen. Unter Punkt 2 lassen sich diese Noten überprüfen, Punkt 3 erlaubt Ihnen, sich ein komponiertes Lied anzuhören und der Punkt 4 wurde vorher schon angedeutet: Von jeder selbstkomponierten Melodie lassen sich hiermit DATAs erzeugen, die in eigenen Programmen wieder verwendet werden können. Mit Punkt 5 kommen Sie wieder ins Hauptmenue.

Darin läßt sich Menuepunkt 2 aufrufen: Daten verändern und neu bearbeiten lassen.

Zu diesen Funktionen bedarf es keiner großen Erklärungen, das Programm ist vorbildlich menuegesteuert und bietet Ihnen in bei jeder Abfrage in Klammern dahinter die kleinste und größtmögliche Zahl für die Parameter-Eingaben.

Sollten Sie mal aus Versehen bei der Dateneingabe die STOP-Taste erwischen, kein Problem. Sie ist außer Gefecht gesetzt, Sie können Ihre bis dahin eingegebenen Daten nicht durch eine unvorhergesehene Beendigung des Programmes verlieren.

Sehr komfortabel gestaltet sich die Wahlmöglichkeit 3 (Diskettenoperationen), die – wie bereits erwähnt – die Musikfiles in der von Ihnen erstellten Art auf Diskette speichert.

Zum Beenden dieses Musik-Utilies drücken Sie im Hauptmenue die Taste „4“.

(A. Füngling/h.b.)

```
10 rem musik genie ==========1
28                                 <ih>
20 rem (p) 07/87 commodore wel
t=                                 <gh>
30 rem ======================
==                                 <kc>
40 rem (c) 07/87 by
==                                 <gf>
50 rem           a. fuengling
==                                 <ej>
60 rem
==                                 <ef>
70 rem version 7.0  40z/ascii
==                                 <ga>
80 rem 128 pc + floppy
==                                 <im>
90 rem ======================
==                                 <no>

95 gosub 60000                     <oc>
100 ro$=chr$(18):rf$=chr$(146)     <lh>
110 printchr$(14):color0,2:col
or5,1:color4,1                     <ji>
120 ifpeek(215)=128thenprint:p
rintro$"schalten sie den 40-ze
ichen modus an ! "rf$:end          <gg>
130 :                              <lm>
140 dimno(500,3)                   <pe>
150 printchr$(19)chr$(19):scnc
lr                                 <ff>
160 char1,0,0,ro$+"M u s i k -
 G e n i e   by    S.A - Soft"
+rf$                               <bc>
170 printchr$(27)+"m"              <pp>
180 char1,0,24,ro$+" - Alexand
er Fuengling Tel 02581/1022 -
"+rf$                              <no>
190 window0,1,39,23                <im>
200 printchr$(27)+"l"              <bm>
210 rem ********************       <kj>
220 rem *** hauptmenue    ***      <gd>
230 rem ********************       <ln>
240 scnclr                         <ni>
250 char1,10,0,"*** Hauptmenue
 ***"                              <nm>

260 char1,0,2,"Gespeicherte No
ten :":printno;"Speicher";fre(
1)                                 <kn>
270 char1,0,5,">>1<< Neue Musi
kdatei einrichten"                 <hi>
280 char1,0,7,">>2<< Musikdate
n verarbeiten"                     <cl>
290 char1,0,9,">>3<< Disketten
operationen"                       <pl>
300 char1,0,11,">>4<< Programm
 beenden"                          <jg>
310 char1,0,17,"Eingabe :"         <en>
320 char1,0,18,"           "+z2
$:poke208,0                        <ne>
330 getkeya$                       <ek>
340 ifa$<"1"ora$>"4"then330        <oi>
350 char1,10,17,a$                 <cc>
360 on val(a$) goto 400,1100,2
310,2230                           <gf>
370 rem ******************         <ca>
380 rem * neue musikdatei *        <hn>
390 rem ******************         <de>
400 scnclr:print:print"Soundpa
rameter (j/n) ?";:getkeya$:ifa
$<>"j"thenwp=1                     <cf>
410 ifa$="j"then begin             <ii>
420 fori=1to 3:scnclr:char1,0,
0,"Parameter fuer Stimme ":pri
nti                                <hm>
430 char1,0,2,"Richtung Maxfre
q Stufe Welle Impulsbrt."          <nc>
440 char1,0,4,z2$+"         "+z
2$+z2$+z2$+z2$+z2$+"    "+z2$+z
2$+z2$+z2$+z2$+" "+z2$+"      "
+z2$+z2$+z2$+z2$                   <ga>
450 char1,0,3,"":q1=1:gosub830
:                                  <dk>
460 ify1$=chr$(13)thengoto480:
else sp(i,1)=val(y2$)              <ca>
470 ifsp(i,1)<0orsp(i,1)>2then
char1,0,3,chr$(27)+"q":goto450     <mh>
480 char1,9,3,"":q1=5:gosub830     <cl>
490 ify1$=chr$(13)thengoto510:
else sp(i,2)=val(y2$)              <dj>
500 ifsp(i,2)<0orsp(i,2)>65535
thenchar1,9,3,chr$(27)+"q":got
o480                               <mj>
510 char1,17,3,"":q1=5:gosub83
0                                  <hi>
520 ify1$=chr$(13)thengoto540:
else sp(i,3)=val(y2$)              <fm>
530 ifsp(i,3)<0orsp(i,3)>32767
thenchar1,17,3,chr$(27)+"q":go
to510                              <be>
540 char1,23,3,"":q1=1:gosub83
0                                  <ja>
550 ify1$=chr$(13)thengoto570:
else sp(i,4)=val(y2$)              <ho>
560 ifsp(i,4)<0orsp(i,4)>3then
char1,23,3,chr$(27)+"q":goto54
0                                  <fo>
570 char1,29,3,"":q1=4:gosub83
0                                  <lh>
580 ify1$=chr$(13)thengoto630:
else sp(i,5)=val(y2$)              <jk>
590 ifsp(i,5)<0orsp(i,5)>4095t
henchar1,29,3,chr$(27)+"q":got
o570                               <cg>
600 char1,0,5,"Probefrequenz :
 "+chr$(27)+"q":q1=5:gosub830:
fr=val(y2$):ify1$=chr$(13)then
```

```
630                                  <nd>
610 char1,0,6,"Probedauer       :
 "+chr$(27)+"q":q1=5:gosub830:
ify1$=chr$(13)then630                <jg>
620 sound1,fr,val(y2$),sp(i,1)
,sp(i,2),sp(i,3),sp(i,4),sp(i,
5)                                   <bg>
630 poke208,0:char1,0,8,"Alles
 soweit richtig (j/n) ?":getke
ya$                                  <og>
640 ifa$="n"thenchar1,0,3,chr$
(27)+"q":goto450                     <hb>
650 nexti:bend                       <ki>
660 ifwp=1thenbegin                  <gi>
670 char1,0,4,"In diesem Fall
nimmt das Programm":                 <mg>
680 char1,0,5,"die Normal - Pa
rameter."                            <hp>
690 fori=1to3:sp(i,4)=2:sp(i,5
)=2048:nexti:sleep1                  <me>
700 bend                             <nf>
710 scnclr                           <la>
720 char1,0,0,"Ihre Parameter
sehen nun wie folgt aus :"           <lm>
730 char1,0,2,"SOUND1,b,c":for
i=1to5:print","sp(1,i);:nexti        <al>
740 char1,0,4,"SOUND2,b,c":for
i=1to5:print","sp(2,i);:nexti        <bj>
750 char1,0,6,"SOUND3,b,c":for
i=1to5:print","sp(3,i);:nexti        <ch>
760 ifno>0thenprint:print:prin
t"Wollen Sie die alten Noten l
oeschen":print"(j/n) ?":getkey
a$:elsechar1,0,9,"Taste.":getk
eyw$                                 <pa>
770 ifa$="j"thenno=0                 <fl>
780 goto240                          <co>
790 rem  *****************           <hc>
800 rem  * input-eingabe *           <ko>
810 rem  *****************           <ig>
820 ez$="qwertyuiopasdfghjklzx
cvbnmQWERTYUIOPASDFGHJKLZXCVBN
M*@?"                                <fh>
830 ez$=ez$+"1234567890"             <he>
840 iflen(ez$)>70thenez$="":go
to830                                <id>
850 gosub860:eg$=y2$:return          <jd>
860 q9=peek(2598):poke2598,0:q
4=0:y2$="":poke2599,0                <bb>
870 getkeyy1$                        <lb>
880 ify1$=chr$(13)then980            <cc>
890 ify1$=chr$(10)then980            <cj>
900 ify1$=chr$(20)then1030           <fh>
910 ifq4=q1thenprintchr$(7);:g
oto870                               <ab>
920 q5=instr(ez$,y1$):ifq5=0th
enprintchr$(7);:goto870              <oo>
930 poke2600,2                       <gc>
```

```
940 ifpeek(2598)then940              <gj>
950 printy1$;:y2$=y2$+y1$:q4=q
4+1:goto870                          <ba>
960 :                                <pn>
970 :                                <ah>
980 poke2600,2                       <je>
990 ifpeek(2598)then990              <ka>
1000 poke2599,1:poke2598,q9          <bh>
1010 ify2$<>""theny1$=""             <cc>
1020 return                          <in>
1030 ifq4=0thenprintchr$(7);:g
oto870                               <ci>
1040 poke2600,2                      <nb>
1050 ifpeek(2598)then1050            <ab>
1060 printchr$(20);:q4=q4-1:y2
$=left$(y2$,q4):goto870              <lo>
1070 rem ***************             <an>
1080 rem * nebenmenue *              <ab>
1090 rem ***************             <cb>
1100 scnclr                          <di>
1110 char1,10,0,"*** Nebenmenu
e ***"                               <cc>
1120 char1,0,2,"Gespeicherte N
oten :":printno;"Speicher";fre
(1)                                  <am>
1130 char1,0,5,">>1<< Noten ei
ngeben"                              <bc>
1140 char1,0,7,">>2<< Noten ue
berpruefen"                          <gk>
1150 char1,0,9,">>3<< Lied spi
elen"                                <jj>
1160 char1,0,11,">>4<< Datas e
rstellen"                            <lk>
1170 char1,0,13,">>5<< Nebenme
nue verlassen"                       <eb>
1180 char1,0,16,"Eingabe"            <hl>
1190 char1,0,17,"           "+z2$
:poke208,0                           <dm>
1200 getkeya$:ifa$<"1"ora$>"5"
then1200                             <ko>
1210 char1,8,16,a$                   <fh>
1220 onval(a$)goto1260,1420,17
90,1880,240                          <nf>
1230 rem *****************           <ff>
1240 rem * noten eingeben *          <ia>
1250 rem *****************           <gj>
1260 dowhilenotno=500:scnclr:n
o=no+1                               <ih>
1270 char1,0,1,str$(no)+". Mus
ikdate eingeben :"                   <fm>
1280 char1,0,3,"Stimme...(1-3)
.....: "+chr$(27)+"q":q1=1:gos
ub830:ifval(y2$)<1orval(y2$)>3
then1280                             <ma>
1290 no(no,1)=val(y2$)               <go>
1300 char1,0,4,"Frequenz.(0-65
535).: "+chr$(27)+"q":q1=5:gos
ub830:ifval(y2$)<0orval(y2$)>6
```

```
5535then1300                              <ah>
1310 no(no,2)=val(y2$)                    <id>
1320 char1,0,5,"Dauer....(0-32
767).: "+chr$(27)+"q":q1=5:gos
ub830:ifval(y2$)<0orval(y2$)>3
2767then1320                              <kl>
1330 no(no,3)=val(y2$)                    <ji>
1340 char1,0,8,ro$+"LINE FEED"
+rf$+" - quittiert/"+ro$+"TAST
E"+rf$+" - fortfahren"                    <hg>
1350 getkeya$                             <ek>
1360 ifa$=chr$(10)thenexit                <ae>
1370 loop                                 <el>
1380 goto1100                             <le>
1390 rem ******************               <pg>
1400 rem * daten anzeigen *               <bn>
1410 rem ******************               <ak>
1420 ifno=0thenscnclr:char1,0,
3,"Keine Daten vorhanden !":ch
ar1,28,5,"Taste.":getkeya$:got
o1100                                     <nn>
1430 do:n=n+1:ifn>no then n=1             <cc>
1440 scnclr                               <in>
1450 char1,0,1,ro$+"LINE FEED"
+rf$+"-quittieren / "+ro$+"v"+
rf$+"-verandern"                          <ch>
1460 char1,0,5,"--------------
---------------------------"              <pp>
1470 char1,0,9,"--------------
---------------------------"              <an>
1480 soundno(n,1),no(n,2),no(n
,3),sp(no(n,1),1),sp(no(n,1),2
),sp(no(n,1),3),sp(no(n,1),4),
sp(no(n,1),5)                             <ef>
1490 char1,0,6,str$(n)+"> SOUN
D":printno(n,1)","no(n,2)","no
(n,3)                                     <ol>
1500 ifno(n,1)<no(n+1,1)thenbe
gin                                       <oj>
1510 n=n+1:soundno(n,1),no(n,2
),no(n,3),sp(no(n,1),1),sp(no(
n,1),2),sp(no(n,1),3),sp(no(n,
1),4),sp(no(n,1),5)                       <mg>
1520 char1,0,7,str$(n)+"> SOUN
D":printno(n,1)","no(n,2)","no
(n,3)                                     <ak>
1530 ifno(n,1)<no(n+1,1)thenbe
gin                                       <ah>
1540 n=n+1:soundno(n,1),no(n,2
),no(n,3),sp(no(n,1),1),sp(no(
n,1),2),sp(no(n,1),3),sp(no(n,
1),4),sp(no(n,1),5)                       <of>
1550 char1,0,8,str$(n)+"> SOUN
D":printno(n,1)","no(n,2)","no
(n,3)                                     <ck>
1560 bend:bend                            <ig>
1570 getkeya$                             <ch>
1580 ifa$=chr$(10)thenexit                <ob>
1590 ifa$="v"thenbegin                    <df>
1600 char1,0,13,"Welche Nummer
 (1-"+str$(no)+") .. "+chr$(27
)+"q":q1=3:gosub830                       <ab>
1610 ify1$=chr$(13)thengoto171
0                                         <la>
1620 ifval(y2$)>no orval(y2$)<
1then1600:else wa=val(y2$)                <nn>
1630 char1,0,14,"Welches Feld
(1-3) ... "+chr$(27)+"q":q1=1:
gosub830                                  <mp>
1640 ify2$<"1"ory2$>"3"then163
0:elsewb=val(y2$)                         <oh>
1650 char1,0,15,"Feld "+str$(w
b)+" ... "+chr$(27)+"q":ify2$<
"2"thenq1=1:elseq1=5                      <nn>
1660 gosub830                             <kk>
1670 ifwb=1andy2$<"1"ory2$>"3"
thengoto1650                              <jm>
1680 ifwb=2andy2$<"0"ory2$>"65
535"thengoto1650                          <hl>
1690 ifwb=3andy2$<"0"ory2$>"32
767"thengoto1650                          <ih>
1700 no(wa,wb)=val(y2$)                   <gm>
1710 char1,0,17,"Noch etwas ve
raendern (j/n) ?":getkeya$                <ml>
1720 ifa$="j"thenprintchr$(27)
+"p":goto1600:elsechar1,0,13,c
hr$(27)+"@"                               <fo>
1730 bend                                 <np>
1740 loop                                 <lo>
1750 goto1100                             <ch>
1760 rem *****************                <bf>
1770 rem * lied spielen *                 <ab>
1780 rem *****************                <cj>
1790 ifno=0thenscnclr:char1,0,
3,"Keine Daten vorhanden !":ch
ar1,28,5,"Taste.":getkeya$:got
o1100                                     <fa>
1800 forn=1tono                           <ac>
1810 soundno(n,1),no(n,2),no(n
,3),sp(no(n,1),1),sp(no(n,1),2
),sp(no(n,1),3),sp(no(n,1),4),
sp(no(n,1),5)                             <jb>
1820 nextn                                <pd>
1830 sleep1                               <gh>
1840 goto1100                             <ic>
1850 rem *******************              <oo>
1860 rem * datas erstellen *              <ej>
1870 rem *******************              <ac>
1880 scnclr:ifno=0thenchar1,0,
2,"Es gibt keine Daten !":char
1,20,4,"Taste":getkeya$:goto11
00                                        <im>
1890 char1,0,1,ro$+"Datas este
llen :"+rf$                               <pl>
1900 char1,0,3,"Das Programm c
riert nun die Datas Ihres"                <ga>
```

```
1910 char1,0,4,"Liedes. Wenn e
s damit fertig ist,"                  <lk>
1920 char1,0,5,"loescht es sic
h selbst, d.h. es "                   <ap>
1930 char1,0,6,"existieren dan
n nur noch Ihre Datas !!"             <of>
1940 char1,0,8,"Sind Sie siche
r, dass Sie nun beginnen"             <ji>
1950 char1,0,9,"wollen (j/n) ?
"                                     <ap>
1960 getkeya$                         <ko>
1970 ifa$<>"j"then1100                <ae>
1980 scnclr                           <kl>
1990 print"10000 fori=1to"no":
reada,b,c"                            <db>
2000 print"10010 onagosub10030
,10040,10050"                         <dl>
2010 print"10020 nexti:end            <ci>
2020 a$="10030 sound1,b,c,"+st
r$(sp(1,1))+","+str$(sp(1,2))+
","+str$(sp(1,3))+","+str$(sp(
1,4))+","+str$(sp(1,5))+":retu
rn":printa$                           <pe>
2030 a$="10040 sound2,b,c,"+st
r$(sp(2,1))+","+str$(sp(2,2))+
","+str$(sp(2,3))+","+str$(sp(
2,4))+","+str$(sp(2,5))+":retu
rn":printa$                           <af>
2040 a$="10050 sound3,b,c,"+st
r$(sp(3,1))+","+str$(sp(3,2))+
","+str$(sp(3,3))+","+str$(sp(
3,4))+","+str$(sp(3,5))+":retu
rn":printa$                           <bg>
2050 print"goto2000": rem ****
* 1 *****                             <ia>
2060 poke842,19:fori=1to7:poke
842+i,13:nexti                        <ji>
2070 poke208,8                        <ld>
2080 end                              <ki>
2090 zh=10090:n=0: rem ***** 1
 ******                               <lc>
2100 scnclr:zh=zh+10:n=n+1:ifn
=no+1then2160:rem ***** 2 ****
*                                     <ao>
2110 printzh"data"no(n,1)","no
(n,2)","no(n,3)                       <in>
2120 print"goto2010":rem *****
 2 *****                              <mi>
2130 poke842,19:poke843,13:pok
e844,13:poke208,3                     <pa>
2140 end                              <oe>
2150 scnclr                           <fg>
2160 scnclr                           <ga>
2170 print"delete-9999":print:
print:print:print"Speichern Si
e sich Ihre Datazeilen ab !!"         <ck>
2180 poke842,19:poke843,13:pok
e208,2                                <ca>
2190 end                              <bg>
2200 rem *********************        <hh>
2210 rem * programm beenden *         <od>
2220 rem *********************        <il>
2230 scnclr:printchr$(19)chr$(
19)                                   <hn>
2240 char1,0,3,"Tschuess, bis
zum naechsten Mal !"                  <ba>
2250 char1,0,5,""                     <ba>
2260 end                              <fm>
2270 rem * programm beenden *         <bp>
2280 rem *********************        <mh>
2290 rem * disk-operationen *         <hj>
2300 rem *********************        <nl>
2310 scnclr                           <ph>
2320 print:printro$"Disketten
operationen"rf$                       <lj>
2330 print                            <lm>
2340 print:print">>1<< - Datei
 laden"                               <hk>
2350 print:print">>2<< - Datei
 speichern"                           <mc>
2360 print:print">>3<< - Diske
tte formatieren"                      <fo>
2370 print:print">>4<< - Diske
ttenverzeichnis"                      <mf>
2380 print:print">>5<< - Datei
en umbenennen"                        <ma>
2390 print:print">>6<< - Datei
en loeschen"                          <db>
2400 print:print">>7<< - Menue
 quittieren"                          <hf>
2410 char1,0,19,"Eingabe :"           <il>
2420 char1,0,20,"          "+z
2$:poke208,0                          <aj>
2430 getkeya$                         <ig>
2440 ifa$<"1"ora$>"7"then2430         <fk>
2450 char1,10,19,a$:a=val(a$)         <ai>
2460 onagoto2500,2720,2870,302
0,3090,3220,240                       <fj>
2470 rem *****************            <ai>
2480 rem * dateien laden *            <pk>
2490 rem *****************            <bm>
2500 scnclr                           <lf>
2510 print:printro$"Datei lade
n"rf$:q1=16                           <ci>
2520 print:print"Name der Date
i : ";:gosub820:q1=80                 <ka>
2530 ify1$=chr$(13)then2310           <lp>
2540 dopen#1,(y2$)                    <ia>
2550 gosub3290:iff$="fehler"th
enf$="":dclose#1:goto2500             <gb>
2560 input#1,no:fori=1to3:foro
=1to5:input#1,sp(i,o):nexto,i         <fp>
2570 fori=1tono:foro=1to3             <mg>
2580 input#1,no(i,o)                  <lb>
2590 nexto,i:close1:ifyt=1then
yt=0:return                           <mm>
```

```
2600 scnclr                          <bk>
2610 print:printro$"Datei lade
n"rf$                                <li>
2620 print:print                     <lc>
2630 print"Name         : "y2$       <lh>
2640 print:print"Daten       : "
no                                   <mn>
2650 print                           <pn>
2660 print"Diskstatus: "ds$          <mp>
2670 print:print"Taste ";:getk
eya$                                 <jn>
2680 goto240                         <kb>
2690 rem *********************       <in>
2700 rem * dateien speichern *       <be>
2710 rem *********************       <kb>
2720 scnclr                          <jc>
2730 print:printro$"Dateien sp
eichern"rf$                          <aa>
2740 print:print"Name der Date
i: ";:q1=16:poke208,0:gosub820
:ify1$=chr$(13)thengoto2310          <nk>
2750 print:print"Bitte haben S
ie etwas Geduld ..."                 <hc>
2760 dopen#1,(y2$),w:gosub3290
:iff$="fehler"thengoto2720           <el>
2770 print#1:print#1,nochr$(13
):fori=1to3:foro=1to5:print#1,
sp(i,o)chr$(13):nexto,i              <fe>
2780 fory=1tono:foru=1to3            <ko>
2790 print#1,no(y,u)chr$(13)         <cj>
2800 nextu,y:dclose#1                <pb>
2810 print:print"Diskstatus :
"ds$                                 <di>
2820 print:print"Taste ";:getk
eya$                                 <de>
2830 goto240                         <di>
2840 rem ***************             <ci>
2850 rem * formatieren *             <em>
2860 rem ***************             <dm>
2870 scnclr                          <cj>
2880 print:printro$"Diskette f
ormatieren"rf$                       <jf>
2890 print:print"Diskettenstat
ion fertigmachen !!"                 <ab>
2900 poke208,0:q1=16                 <mb>
2910 print:print"Name der Disk
ette : ";:gosub820                   <ei>
2920 ify1$=chr$(13)then2310          <eh>
2930 na$=z2$                         <jc>
2940 print:print:print"Druecke
n Sie eine Taste, dann kann es
 losgehen ! ";:getkeya$              <ff>
2950 header(na$),d0,isa              <cp>
2960 print:printds$                  <mc>
2970 print:print"Taste ";:getk
eya$                                 <mk>
2980 goto2310                        <po>
2990 rem *************               <gk>
3000 rem * directory *               <fl>
3010 rem ******+*****                <hp>
3020 scnclr                          <lp>
3030 directory                       <mp>
3040 poke208,0:print:print"Tas
te ";:getkeya$                       <nh>
3050 goto2310                        <ee>
3060 rem *********************       <aa>
3070 rem *dateien umbenennen *       <nb>
3080 rem *********************       <bf>
3090 scnclr:print:printro$"Dat
eien umbenennen"rf$                  <ng>
3100 print:print"Alter Name :
";:gosub820                          <eh>
3110 ify1$=chr$(13)then2310          <ag>
3120 al$=y2$                         <eo>
3130 print:print"Neuer Name :
";:gosub820                          <gm>
3140 ne$=y2$                         <gi>
3150 rename(al$)to(ne$)              <pn>
3160 print:print"Diskstatus :
"ds$                                 <ji>
3170 print:print"Taste ";:getk
eya$                                 <jd>
3180 goto2310                        <mh>
3190 rem ********************        <fj>
3200 rem * dateien loeschen *        <lk>
3210 rem ********************        <gn>
3220 scnclr                          <ii>
3230 print:printro$"Dateien lo
eschen"rf$:q1=16                     <hl>
3240 print:print"Zu loeschende
n Datei : ";:gosub820                <hi>
3250 ify1$=chr$(13)then2310          <jc>
3260 scratch(y2$):q1=80              <ja>
3270 print:print"Diskstatus :
"ds$                                 <ag>
3280 print:print:print"Taste "
;:getkeya$:goto2310                  <fn>
3290 ifds>0thenf$="fehler":els
e3420                                <ho>
3300 ifds>19thenprint:print"Di
skfehlermeldung : ";ds$              <cb>
3310 ifds=20thenprint"Blockken
nsatz nicht gefunden"                <mp>
3320 ifds=21thenprint"Kein Syn
chronisationszeichen"                <mk>
3330 ifds=22thenprint"Datenblo
ck nich verf"+zs$+"gbar"             <pm>
3340 ifds=23thenprint"Pr"+zs$+
"fsummenfehler im Datenblock"        <bp>
3350 ifds=26thenprint"Diskette
 hat einen Schreibschutz"            <gn>
3360 ifds=30thenprint"Fehler i
m Filenamen"                         <md>
3370 ifds=62thenprint"Datei ni
cht gefunden"                        <ka>
3380 ifds=63thenprint"Datei is
```

```
t schon vorhanden"                      <pj>
3390 ifds=72thenprint"Diskette
 ist voll."                             <ee>
3400 ifds=74thenprint"Das Lauf
werk ist nich bereit"                   <bo>
3410 getkeya$                           <fo>
3420 return                             <ph>
60000 rem nachspann ==========
=====                                   <ag>
60010 rem *** zeichensatz/grap
hik *                                   <ai>
60020 z2$=chr$(163):zs$=chr$(1
89)                                     <oj>
60030 return                            <pg>
```

C128 - ROM IM ÜBERBLICK

Wo liegt was im C128-ROM? Alle Routinen, die zu nutzen vorteilhaft ist, haben wir zusammengetragen und alphabetisch nach deren Kurzbezeichnung aufgelistet – als Wegweiser durch das C128-Kernel.

„Wo ist denn nur diese ROM-Unterroutine??? Gestern lag sie doch noch bei $BC44! Ich hatte sie mir doch aufgeschrieben...“. Die Freßzettelpinselei und das Herumgesuche nach 128er Kernel-Routinen hat ein Ende. Es genügt fortan, diesen Artikel zum Computer zu legen und schon haben die Freaks der Maschinensprache alles in der Hand. Mehrfachnennungen sind, durch verschiedenen Kurzbezeichnungen, beabsichtigt. So ist eine bestimmte Routine leichter zu finden.

Sven Faulhaber ■

$E43E: ACPTR-Routine
$C320: ASCII in Bildschirm-Code umwandeln
$B211: ASCII-Ausgabe von 8/16-Byte
$C320: ASCII-Wert in Bildschirm-Codewert umrechnen
$F9FB: ASCII-Zeichen aus Akkumulatorinhalt aufbereiten
$B8D2: ASCII-Zeichen aus Akkumulatorinhalt errechnen
$EA8F: Abfrage der Datasetten-STOP-Taste
$C3F4: Abfragen der Commodore-Taste
$B641: Adresse eines Branch-Befehls errechnen
$BOC5: Adresse eines Monitorbefehls ermitteln
$B8A5: Akkumulator als ASCII ausgeben und zum Zeilenanfang gehen
$B11A: Akkumulator aus beliebiger Bank laden
$B12A: Akkumulator in beliebige Speicherbank schreiben
$CDCA: Akkumulator in das VCR-Data-Register schreiben
$B13D: Akkumulator mit b beliebiger Speicherstelle vergleichen
$B8D2: Akkumulator wandeln in X-Register-ASCII-Zeichen low, Akkumulator-ASCII-Zeichen high
$B8C2: Akkumulatorinhalt als 2 ASCII-Zeichen (high/low) ausgeben
$F9FB: Akkumulatorinhalt in ASCII-Zeichen aufbereiten
$B8D2: Akkumulatorinhalt in ASCII-Zeichen umwandeln. Ergebnisse in Akkumulator und X-Register
$CC34: Akkumulatorinhalt in X-Farbe ausgeben
$C15C: Aktuelle Adresse des Cursors errechnen
$C72D: Ausgabe auf Bildschirm
$BO50: Ausgabe der Register und Prozessorwerte
$C9C1: Ausgabe einer ESC-Sequenz
$CC27: Ausgabe eines Leerzeichen an der Cursorposition
$B5F2: Ausgabe von drei Leerzeichen
$B3C4: Ausgabe von ERROR
$B8A5: Ausgabe: Akkumulator als ASCII-Zeichen und zum Zeilenanfang gehen
$B8C2: Ausgabe: Akkumulatorinhalt als zwei Buchstaben (high/low) ausgeben
$BA5D: Ausgabe: OPA
$C5D7: Auswertung der Tasturergebnisse
$CAE9: Auto-Insert ausschalten
$CAEC: Auto-Insert einschalten
$EF06: BASIN
$F934: BOOT
$EF79: BSOUT
$BOE3: BASIC-Warmstart
$C98E: Bell
$CE8E: 80-Zeichen-Bildschirm initialisieren
$CA24: Bildschirm als Fenster definieren
$CEOC: Bildschirm auf 80 Zeichen schalten
$CE74: Bildschirm für 40 Zeichen initialisieren
$C72D: Bildschirmausgabe
$CD2E: Bildschirmausgabe 40/80-Zeichen tauschen
$CC5B: Bildschirmausgabe neu organisieren
$CB3E: Bildschirmausgabe reverse
$C6E7: Blinkkontrolle des Cursors
$B641: Branch-Befehlsadresse feststellen
$E3E2: Byte an IEC-Bus senden
$E224: C128MODE
$E795: CHKIN
$F127: CHKIN-Auswertung für IEC-Bus
$CO7B: CINT
$E503: CIOUT
$E723: CKOUT
$F16D: CKOUT für IEC-Bus
$F222: CLALL
$F188: CLOSE
$C142: CLR/HOME
$F226: CLRCH
$F7E3: CMPFAR
$B13D: CMPVEC
$C76F: Carriage Return
$BB03: Catalog
$C8A6: Shift-Commodore-Taste zulassen
$E545: Clock Signal high
$E54E: Clock Signal low
$C3F4: Commodore-Taste abfragen
$CD57: Cursor
$C150: Cursor Home im Fenster
$CB51: Cursor ans Zeilenende bewegen
$C33E: Cursor ans Zeilenende setzen
$C875: Cursor eins nach links bewegen
$C867: Cursor eins nach oben bewegen
$C854: Cursor eins nach rechts bewegen
$C85A: Cursor eins nach unten bewegen
$C91B: Cursor nach links bewegen und Zeichen löschen
$C94F: Cursor zum Tabulatoren fahren
$CBOA: Cursorblinken ausschalten
$CB20: Cursorblinken einschalten
$C6E7: Cursor-Blinkgeschwindigkeits-Kontrolle
$CC6A: Cursorposition lesen
$CC6A: Cursorposition setzen
$CC1E: Cursorpositionen nach $DE/$DF der Nullseite kopieren
$CC34: DISPLAY
$F7A5: DMA-CALL
$E9C8: Datasettentastendruck abwarten
$EA1C: Daten auf Kassette schreiben
$E9F2: Daten von Kassette lesen
$E919: Datenblock auf Kassette schreiben
$C3DC: Delete einer Zeile
$B90E: Differenz aus OP1-OP3 in OP1 speichern
$BB03: Directroy-Routine
$B3C4: ERROR-Meldung ausgeben
$CAEC: ESC-Routine: A-Auto-Insert einschalten
$CA15: ESC-Routine: B-Unteren Fensterrand setzen
$CAE9: ESC-Routine: C-Auto-Insert aus
$CA51: ESC-Routine: D-Laufende Zeile löschen
$CBOA: ESC-Routine: E-Cursorblinken abschalten
$CB20: ESC-Routine: F-Cursorblinken einschalten
$CB36: ESC-Routine: G-Signalton einschalten
$CB39: ESC-Routine: H-Signalton verhindern
$CA3C: ESC-Routine: I-Zeile einfügen
$CBBO: ESC-Routine: J-Cursor zum Zeilenanfang bewegen
$CB51: ESC-Routine: K-Cursor ans Zeilenende bewegen
$CAE1: ESC-Routine: L-Scrollen o.k.

$CAE4: ESC-Routine: M-Scroll-Stop
$CB47: ESC-Routine: N-RVS im 80-Zeichenmodus ausschalten
$C77C: ESC-Routine: O-Insert, RVS- und Qoute-Modi deaktivieren
$CB3E: ESC-Routine: R-Reverser Bildschirm
$CA13: ESC-Routine: T-Oberen Fensterrand setzen
$CA9E: ESC-Routine: Text ab Cursor in Zeile löschen
$CAFD: ESC-Routine: U-Underline Cursor im 80-Zeichen-Modus
$CABB: ESC-Routine: V-aufwärts scrollen
$CAC9: ESC-Routine: W-abwärts scrollen
$CD28: ESC-Routine: X-Umschalten zwischen 40- und 80-Zeichen-Darstellung
$C982: ESC-Routine: Y-Tabulatoren setzen
$C97F: ESC-Routine: Z-Tabulatoren löschen
$C258: Eingabezeile einlesen
$C258: Einlesen einer Eingabezeile
$CB65: Einlesen von Farbe und Zeichen des Cursorstandpunktes
$EF48: Einzelne Zeichen
$C96C: Ermitteln der Tabulatorposition
$C9C1: Escape
$E5D6: FAST an seriellen Bus senden
$F7C9: FETCH
$F7AE: FETCH
$B11A: FETVEC Fetch Vector
$C8D5: FLASH-Modus setzen
$E5F8: FSTMOD
$C7E5: Farbe setzen im 40-Zeichen-Modus
$C7EC: Farbe setzen im 80-Zeichen-Modus
$CA1B: Fenster initialisieren
$C142: Fenster löschen
$CA14: Fenstergrenzen definieren
$CA13: Fensterrand oben setzen
$CA15: Fensterrand unten setzen
$C40D: Fensterzeile kopieren
$C8CD: Flash-Modus zurücksetzen
$CCA2: Funktionstasten definieren
$E7CE: GET aus RS232-Puffer
$E7E4: GET von RS232-Schnittstelle
$F7EC: GETCFG
$EEEB: GETIN
$E24B: G064MODE
$C150: HOME im Fenster
$C8B3: Home-Taste testen
$EF5C: IEC Bus auslesen
$E38C: IEC-Bus-Vorbereitung
$CEOC: INIT 80
$C8E3: INSERT ausführen
$F781: IOBASE
$E109: IOINIT
$C234: IRQ-Buffer auslesen
$C207: IRQ-Register setzen
$C194: IRQ-Routine
$FA65: IRQ-Routine
$C7E5: Initialisieren der Farbe im 40-Zeichen-Modus
$C7EC: Initialisieren der Farbe im 80-Zeichen-Modus
$CA1B: Initialisieren der Fenstergröße
$CA14: Initialisieren der Fenstergrößen
$C207: Initialisieren der IRQ-Register
$E1DC: Initialisieren der VDC-Register
$C880: Initialisieren des zweiten Zeichensatzes
$CE74: Initialisieren des 40-Zeichen-Bildschirms
$CE8B: Initialisieren des 80-Zeichen-Bildschirms
$C961: Initialisieren des Tabulators
$C1DF: Initialisieren des Text-Modus
$CA13: Initialisieren des oberen Fensterrandes
$CO7B: Initialisieren von Editor und Screen
$EO4B: Initialisierungswerte des MMU
$C37C: Insert einer Zeile
$CAEA: Insert-Flag manipulieren
$B1C9: Insert-Modus zurücksetzen
$F82B: JSRFAR
$FC87: KEY
$F867: Kaltstart
$EE57: Kassettenbetrieb beenden
$E8DO: Kassettendrivehandling
$E43E: Kernal-Routine: ACPTR
$C98E: Klingeln
$CEOC: Kopiere Zeichensatz in VDC-RAM
$B901: Kopieren des OP1-Inhaltes nach OP3
$C40D: Kopieren einer Fensterzeile
$C436: Kopieren einer Zeile im 80-Zeichen-Modus
$E343: LISTN
$F79D: LKUPLA
$F786: LKUPSA
$F265: LOAD (Diskette)
$F27B: LOAD (IEC-Bus)
$C258: LOOP4
$C29B: LOOP5
$C234: LP2
$CC27: Leerzeichen an Cursorposition ausgeben
$FC87: Lesen eines Zeichens von Tastatur
$C9B1: Line Feed durchführen
$CA9E: Lösche Zeile ab Cursor
$C980: Löschen der Tabulatoren
$C93D: Löschen des Zeichens unter dem Cursor
$C4A5: Löschen einer 40-Zeichen-Zeile
$C4C0: Löschen einer 80-Zeichen-Zeile
$F772: MEMBOT
$F763: MEMTOP
$E04B: MMU-Tabelle
$BO3A: MONITOR-Textmeldung ausgeben
$C40D: MOVLIN
$BO14: Monitor neu initialisieren
$BO50: Monitorbefehl: Register
$BOE3: Monitorbefehl: X (exit)
$B1DF: Monitorbefehl: Jump
$B194: Minitorbefehl: ; Modify Register
$B1AB: Monitorbefehl: > Modify Memory
$B406: Monitorbefehl: Assemble
$B231: Monitorbefehl: Compare
$B599: Monitorbefehl: Disassemble
$BA90: Monitorbefehl: Diskkommando
$B3DB: Monitorbefehl: Fill
$B1D6: Monitorbefehl: Go to
$B2CE: Monitorbefehl: Hunt
$B337: Monitorbefehl: Load
$B152: Monitorbefehl: Memory Display
$B337: Monitorbefehl: Save
$B234: Monitorbefehl: Transfer
$B337: Monitorbefehl: Verify
$BOC5: Monitorbefehlsadresse ermitteln
$B003: Monitoreinsprung durch Break
$B009: Monitoreinsprung mit BREAK-Ausgabe
$B006: Monitoreinsprung per Exmon
$B000: Monitoreinsprung regulär durch Befehl
$E672: NMI der RS232 setzen
$FA40: NMI-Routine
$E805: NMI-Routine der RS232
$C76F: Neue Zeile beginnen
$BA07: OP1-Inhalt in achtstellige Dezimale umwandeln und ab $AAO speichern
$B901: OP1-Inhalt nach OP3 kopieren
$BA5D: OPA angeben
$EFBD: OPEN (Diskette)
$EFFO: OPEN (Kassette)
$F040: OPEN (RS232)
$CCA2: PFKEY
$F867: PHOENIX
$CC6A: PLOT
$FA17: PRIMM
$C72D: PRINT
$E9F8: Programme von Kassette lesen
$B1C9: Quote Modus zurücksetzen
$C2FF: Quote-Flag testen und eventuell setzen
$E093: RAMTAS
$F65E: RDTIM
$F744: READST
$C651: REPEAT
$E000: RESET
$E056: RESTOR
$E26B: ROM-Test
$EFB7: RS232 Ausgabe
$EF67: RS232 Zeichen auslesen
$E6DF: RS232-Empfang bearbeiten
$E5FF: RS232-Senderroutine
$E6D4: RS232-Startbit testen
$F63D: RUN/STOP-Abfrage
$B20B: RVS-Modus setzen
$B1C9: RVS-Modus zurücksetzen
$E9E9: Record&Play-Tastendruck des Recorders abfragen
$BO50: Register ausgeben
$EEBO: Recordermotor ausschalten
$F53E: SAVE
$F5C8: SAVE (Datasette)
$F585: SAVESP
$CC5B: SCRORG
$E4D2: SECND
$F73F: SETBNK
$F738: SETFLS
$F75C: SETMSG
$F731: SETNAM
$F75F: SETTMO
$F7BC: STASH
$B12A: STAVEC
$F66E: STOP
$CD2E: SWAPPER
$CAE4: Scroll Stop
$C3A6: Scrollen aufwärts
$CAE1: Scrollen o.k.
$C60A: Shift-Commodore-Taste prüfen
$CB39: Signalton abschalten
$CB36: Signalton zulassen
$BA07: Speichern des umgewandelten, achtstelligen Dezimalwertes von OP1 ab $=AAO
$E59F: Spritezeiger zurücksetzen
$E573: Systemtakt auf 1 MHZ setzen
$E33B: TALK
$E4EO: TKSA
$B721: Tabelle Mnemonics des 8510
$B707: Tabelle: Adressierungsarten
$B6C3: Tabelle: Adressierungsreferenz
$F67C: Tabelle: Fehlermeldungen
$E105: Tabelle: RAM-Banks

$FA80: Tabelle: Tastaturzeichensätze
$E2F8: Tabelle: VDC-Initialisierung
$E2C7: Tabelle: VIC-Initialisierung
$CE6C: Tabelle: Zweierpotenzen
$BOFC: Tabellenbeginn: Monitor-Befehle
$C96C: Tabulatorpositionen ermitteln
$C94F: Tabulator anspringen
$C961: Tabulator manipulieren
$C980: Tabulatoren löschen
$FC87: Tastaturabfrage
$C5D7: Tastaturergebnisse auswerten
$C55D: Tastaturmatrix prüfen
$F%33: Textausgabe „LOADING/ Verifying"
$F50F: Textausgabe „Searching for ..."
$F908: Textausgabe: BOOTING
$F6C7: Textausgabe: FOR
$F713: Textausgabe: FOUND
$F6BO: Textausgabe: I/O Error
$F6F9: Textausgabe: LOADING
$F71A: Textausgabe: OK
$F6CB: Textausgabe: PRESS PLAY ON TAPE
$F701: Textausgabe: SAVING
$FCBC: Textausgabe: von Kassette lesen SEARCHING
$F709: Textausgabe: Verifying
$BO3A: Textmeldung Monitor ausgeben
$C1DF: Textmodus setzen
$F5F8: UDTIM
$E526: UNLSN
$E515: UNTLK
$C320: Umrechnen der ASCII-Werte in Bildschirmcodes
$C60A: Umschalten zwischen den Zeichensätzen
$C8CE: Unterstreichen ausschalten
$C8CE: Unterstreichen einschalten
$CDD8: VCR-Data-Registerwert in Akkumulator lesen
$E1DC: VDC-Register initialisieren
$E05B: VEKTOR
$E385: Verzögerungsschleife
$CA1B: WINDOW
$BOE3: Warmstart
$B90E: Wert aus OP1-OP3 in OP1 speichern
$B8E7: Zeichen aus Eingabepuffer lesen
$C234: Zeichen aus IRQ-Buffer holen
$C91B: Zeichen links vom Cursor löschen
$C93D: Zeichen unter Cursor löschen
$C29B: Zeichen vom Bildschirm lesen
$CEOC: Zeichensatz ins VDC-RAM kopieren
$C60A: Zeichensatzumschaltung
$C651: Zeichenwiederholung
$C37C: Zeile einfügen
$C436: Zeile im 80-Zeichen-Modus
$C3DC: Zeile löschen
$C4A5: Zeile speziell im 40-Zeichen-Modus löschen
$C4CO: Zeile speziell im 80-Zeichen-Modus löschen
$C363: Zeilenvorschub
$C880: Zweiten Zeichensatz einschalten



Der C128 und sein TEDMON

EIN MONITOR OHNE BILDSCHIRM

Der softwaremäßig verankerte Maschinensprache-Monitor des C128 erlaubt Funktionen wie Floppy-Bedienung und Verschieben ganzer Speicherblöcke, recht komfortable Befehle zur Eingabe, Lesen und Testen von Maschinensprache, was in BASIC nur unter größten Mühen zu realisieren wäre.

Ein Maschinensprache-Monitor verarbeitet mnemotechnische Ausdrücke und Befehle, die er bereits bei der Eingabe in für den Computer verständliche Codezahlen-Werte umwandelt. Er bedient sich hier der sedezimalen Zahlendarstellung, auch Hex-Zahlen genannt. Diese Werte gehen von einer Basis 2 ↑ 4 (= 16) aus, sie werden

WICHTIG: KENNTNIS DER SEDEZIMALZAHLEN

von den Ziffern 0 bis 9 und, statt 10 bis 15, von den Buchstaben A bis F dargestellt. In der Praxis bedeutet das, daß – abgesehen von der äußersten rechten – alle Stellen nach links mit 16 multipliziert werden müssen Nicht nur das, je weiter links die Zahlenwerte stehen, desto höher werden die Potenzwerte, beginnend bei der zweiten Stelle von rechts (= 16 oder 2 ↑ 4). Der nächste Wert heiß also 2 ↑ 12.

Die Produkte dieser Multiplikationen innerhalb einer solchen Hexzahl addiert ergeben dann ihren Dezimal-Wert. Beispiel: Speicherstelle „01FF" hex. 1 x 256 + 15 x 16 + 15 = 511 dez. Dies nur als kurzes Abschweifen in die Hex-Zahlen-Welt, da Sie bei der Arbeit mit Maschinensprache und einem dafür vorgesehenen Monitor ständig damit konfrontiert werden. Übrigens: Solche Sedezimalzahlen werden immer mit einem vorangestellten Dollarzeichen ($) gekennzeichnet.

MODUS AUFRUFEN MIT MONITOR

In den Maschinensprache-Monitor des C128 kommen Sie recht unkompliziert: entweder durch Eingabe der Anweisung „MONITOR" (mit „Return" abschließen) oder durch Druck auf die Funktionstaste 8, die diese Anweisung sofort ausführt. Jetzt ist auf Ihrem Bildschirm die Registeranzeige erschienen, die so aussieht:

```
MONITOR
   PC  SR AC XR YR SP
;FB000 00 00 00 00 F8
```

Von dem munter blinkenden Cursor darunter sollten Sie sich nicht täuschen lassen, das gewohnte BASIC gibt's momentan nicht mehr für Ihren C128. Alle derartigen Befehle oder Eingaben würde er nur mit einem verwunderten Fragezeichen quittieren.

Was nicht heißen soll, daß Sie ab sofort auf Ihr geliebtes BASIC verzich-

ten müssen. In dieses Betriebssystem können Sie jederzeit wieder zurück: Entweder durch die Eingabe von „X“ (=Exit) oder gleichzeitiges Drükken der Run/Stop- und Restore-Taste. Doch zurück zum C128-Monitor.

DIE EINZELNEN REGISTER-WERTE

PC(Program Counter, Programm-Zähler) ist der Zähler, unter dem die Adresse des Befehles steht, der als nächster abgearbeitet werden soll, also in unserem Beispiel „B000“ (45056 dez.) Was aber bedeutet das „F“ vor der Adreßzahl? Im C128-Monitor kann (muß aber nicht) jeder Hexzahl fünfstellig angegeben werden, dann ist diese fünfte links außen die gerade aktuelle Bank-Konfiguration, also hier „15“ (=F).

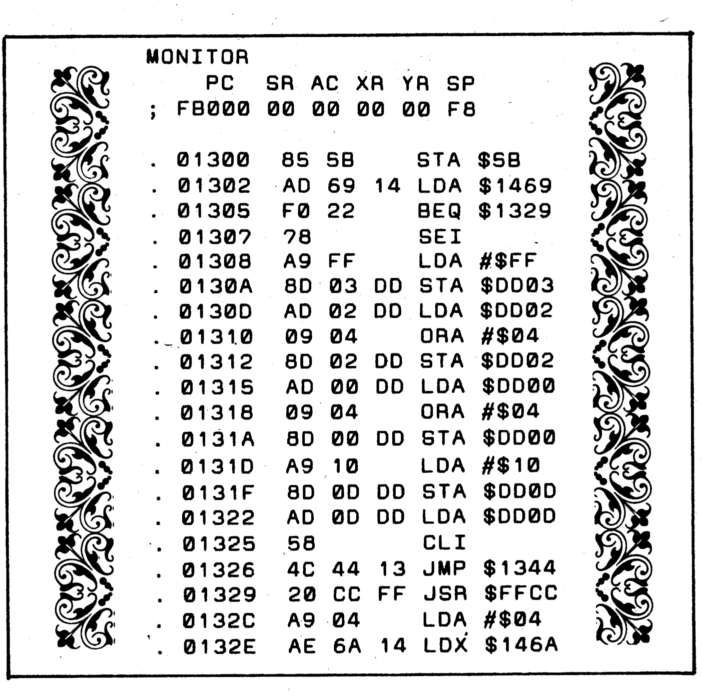
Ohne RAM-Floppy oder sonstige Speichererweiterungen des C128 gibt es sowieso nur vier Möglichkeiten:

0 .... Bank 0 (RAM-Bereich)
1 .... Bank 1 (RAM-Bereich und Variablenspeicher)
E .... Bank 14 (Zeichensatz-ROM)
F .... Bank 15 (Betriebssystem-ROM).

SR (Status Register): Hier wird der Inhalt der Statusregister des Mirkoprozessors 8502 beim C128 angezeigt, darin sind praktisch alle Flags (Flaggen, Merker) zusammengefaßt.

ALLE FLAGGEN GEHISST?

Ein Flag ist ein Bit, das einen ganz bestimmten Zustand des Mikroprozessors Ihres Computers anzeigt. Die weitere Tätig-



```
MONITOR
    PC  SR AC XR YR SP
; FB000 00 00 00 00 F8

. 01300  85 5B     STA $5B
. 01302  AD 69 14  LDA $1469
. 01305  F0 22     BEQ $1329
. 01307  78        SEI
. 01308  A9 FF     LDA #$FF
. 0130A  8D 03 DD  STA $DD03
. 0130D  AD 02 DD  LDA $DD02
. 01310  09 04     ORA #$04
. 01312  8D 02 DD  STA $DD02
. 01315  AD 00 DD  LDA $DD00
. 01318  09 04     ORA #$04
. 0131A  8D 00 DD  STA $DD00
. 0131D  A9 10     LDA #$10
. 0131F  8D 0D DD  STA $DD0D
. 01322  AD 0D DD  LDA $DD0D
. 01325  58        CLI
. 01326  4C 44 13  JMP $1344
. 01329  20 CC FF  JSR $FFCC
. 0132C  A9 04     LDA #$04
. 0132E  AE 6A 14  LDX $146A
```

keit dieses Chips ist immer davon abhängig, ob so ein Flag gesetzt (=1) oder ausgeschaltet ist (=0). Folgende Flags werden – von Fall zu Fall eingesetzt:

Carry-Flag: Findet seinen Funktionsbereich vor allem bei den Rechenvorgängen Addition und Subtraktion. Der Akkumulator verarbeitet nur acht Bit, also einen Höchstwert von 255. Tritt nun bei einem solchen Rechenvorgang ein Übertrag auf, wird das Carry-Flag gesetzt.

Zero-Flag: Wird dann gesetzt, wenn das Ergebnis einer arithmetischen Operation gleich Null ist.

Unterbrechungsverhinderungs-Flag: Ist dieses Flag gehißt, kann keine Unterbrechung durch den Interrupt erfolgen (normalerweise geschieht so etwas sonst jede 1/60 sec).

Dezimal-Flag: Trifft die Auswahl zwischen Binär-Arithmetik und BDC-Dezimal-Arithmetik.

BRK-Flag: Wird gesetzt, wenn ein Maschinenprogramm vom Befehl „BRK“ (Break) beendet wird.

Überlauf-Flag: Kommt bei einer vorzeichenbehafteten Addition oder Subtraktion zum Tragen, wenn der zulässige Bereich von –128 bis +127 überschritten wird. Wäre das nicht der Fall, gäbe es ständig einen Overflow.

Negativ-Flag: Wird verwendet, wenn ein Rechenergebnis ein negatives Vorzeichen hat.

AC (Accumulator): Der aktuelle Inhalt des Akkumulators wird angezeigt. Der Akku ist so ziemlich das wichtigste Register eines Prozessors, vor allem in bezug auf arithmetische Vorgänge. Der Wert einer Speicherstelle kann direkt in den Akku geladen, bearbeitet und wieder in einer anderen abgelegt werden.

XR und *YR (X- und Y-Register):* Zusätzliche Prozessorregister X und Y, beide dienen als Laufvariable für Schleifen. Ebenso werden sie zur Indizierung von Tabellen verwendet.

SP: Hiermit ist der Stapelzeiger (*Stack Pointer*) gemeint, der Bereich, der sich an die Zeropage ab Speicherstelle 256 – 511 ($0100 – $01FF) anschließt. Dieser Speicherplatz wird vor allem zum Zwischenspeichern von Registerwerten verwendet.

Diese Register-Anzeige können Sie innerhalb des Monitors jederzeit mit „R“ aufrufen, was gerade beim Programmieren in Maschinensprache sehr hilfreich sein kann. Die Inhalte der Prozessorregister können jederzeit durch einfaches Überschreiben geändert werden.

DIE MONITOR-ARBEITSBEFEHLE

Wie es sich für einen guten Monitor gehört, stellt auch dieser recht komfortable Befehle zur Verfügung, um bestimmte Speicherbereiche Ihres Computers beschreiben oder programmieren zu können.

A (Assemble): Der wohl wichtigste Befehl zum Programmieren in Maschinensprache. Gehen wir einmal davon aus, Sie sollten ab Adresse 4864 oder $ 1300 ein Maschinensprache-Programm eingeben (das von BASIC aus dann mit einem SYS-Befehl aufgerufen wird), das als ersten Schritt den Wert „11“ in den Akku laden soll:

A 1300 LDA#$OB

Nach dem Drücken von „Return“ nimmt Ihre Eingabezeile jetzt folgende Form an:

A 1300 A9 OB LDA#$OB
A 1302 (hier steht jetzt der Cursor.)

„A“: SAMMELT BYTE

Daran erkennen Sie, daß nach dem „Sammeln“ (Assemblieren) links die entsprechenden Byte angegeben werden, also “A9” (=169) und “OB” (=11). Das bedeutet für den Computer, den Wert “11” in den Akku zu laden. In BASIC hätten wir das per POKE erledigen

müssen, außerdem sind dort nur Dezimalzahlen erlaubt:
POKE 4864,169: POKE 4865,11.
Der Cursor meldet sich wieder mit dem "A"-Befehl und der Speicherstelle, ab der weiter assembliert werden kann, demnach "1302".

```
MONITOR
     PC  SR AC XR YR SP
; FB000 00 00 00 00 F8

>00100 34 35 30 00 30 30 30 30 30 30 00 00 00 00 00 00:
>00110 FF 0E 00 00 00 00 00 FF FF FF FF 00 08 00 00 0F:
>00120 00 00 00 DE B4 02 00 B4 02 DE B4 02 DE B4 02 DE:
>00130 B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4:
>00140 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02:
>00150 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE:
>00160 B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4:
>00170 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02:
>00180 DE B4 02 DE B4 02 DE B4 02 DE B4 B4 02 DE B4 02:
>00190 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE:
>001A0 B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4:
>001B0 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02:

>001C0 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE B4 02 DE:
>001D0 31 C2 7C 31 C2 7C FA 31 3F F7 31 C2 7C 31 C2 7C:
>001E0 FA 00 1F 1F BF F3 CD 31 C2 31 C2 7C FA 41 E4 18:
>001F0 E4 01 0F 24 B1 FC B1 7E B1 F5 4A 4A 37 52 2B 40:
>00200 4D 00 2C 34 2C 30 3A 9D 31 3A FA 00 4E 49 54 4F:
>00210 52 00 52 00 4F 52 00 50 52 47 00 00 FF FF FF FF:
>00220 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF:
>00230 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF:
>00240 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00:
>00250 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00:
>00260 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00:
>00270 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00:
```

„D": DER LIST-BEFEHL DES MONITORS

D (Dissassemble): Dieser Befehl ist vergleichbar mit der PEEK-Funktion oder „LIST" in BASIC. Wenn Sie hier
D 1300
eingeben, so erscheint eine Bildschirmseite lang, das von Ihnen in diesen Speicherbereich mit "A" geschriebene Maschinenprogramm (oder jeder gewünschte Speicherbereich). Unser Beispiel würde dann so aussehen:

```
D1300
D1300 A9 0B LDA#$0B
D1302 00        BRK
D1303 FF        ???
D1304 00   BRK usw.
```

„BRK" und „???" bedeuten nur, daß nichts mehr Vernünftiges im folgenden Speicherbereich steht, er ist schlicht und einfach leer. Der Punkt "." vor den jeweiligen Adresen hat diesselbe Funktion wie "A", Sie könnten also hier jede Befehlseingabe ändern.

M (Memory Dump): Das bedeuter soviel wie „Speicherauszug". Wenn Sie
M 1300
eingeben, so wird ab dieser Speicherstelle – wie an einer Kette aufgereiht – der Inhalt der folgenden Adressen hexadezimal angezeigt.

Vor den Hexdump-Zeilenauf dem Bildschirm erscheint dann eine Spitzklammer nach rechts (ebenfalls ein Monitor-Befehlszeichen). Sie gibt zu verstehen, daß Sie nun ungestraft und ohne Datenverlust diese angezeigten Byte ändern können. Überprüfen läßt sich das am besten an dem am rechten Rand gezeigten Klartext, der praktisch die ASCII-Code-Pendants der eingegebenen Zeichen, revers dargestellt.

JAGT UND FINDET BYTE-FOLGEN: „H"

H („hunt", suchen): Immer wieder kommt es vor, daß Sie in einem gewissen Speicherbereich nach einer Byte-Folge suchen wollen, um sie zu ändern oder sonstwie bearbeiten zu können. Auch hier hilft uns der Monitor mit einer komfortablen Befehlseingabe:
H 1300 1350 A9
Hinter H wird die Anfangs- und Endadresse des zu durchsuchenden Speicherbereichs und die Bytefolge (es kann natürlich auch eines alleine sein) eingegeben. Da das gesuchte Byte unseres Minibeispiels nur in der Adresse „1300" steht, würde diese darunter auch ausgegeben werden. „Hunt" kann aber noch mehr: Falls Sie in einer Tabelle nach einem bestimmten Wort oder Text suchen, so müssen Sie diese Byte-Folge nicht unbedingt hexadezimal vermerken, was vielleicht für den einen oder anderen zu umständlich wäre. Trennen Sie diese Eingabe mit einem Hochkomma " ' " ab und schreiben Sie die zu suchenden Byte im ASCII-Format, etwa:
H 1500 1600 'BYTE
Die folgende Anzeige in der nächsten Zeile gibt Ihnen die Adressen an, wo der in diesem Bereich Wort-String Byte zu finden ist.

T („transform", verschieben): Damit läßt sich ein genau umgrenzter Speicherbereich innerhalb des freien Speichers verschieben. Wenn Sie nun eine kleine Routine aus dem klassischen Maschinensprachebereich des C128 beispielsweise in den BASIC-Speicher ab $4000 (16384) ablegen möchten, so würde das etwa so aussehen:
T 1300 1400 4000
Sie geben damit Anfangs- und Endadresse des zu verschiebenden Bereichs und die Anfangsadresse des neuen Speicherplatzes, an den geschoben werden soll, ein. Die Endadresse muß nicht mit angegeben werden, die rechnet sich der Computer selbst aus. Die Originalroutine ab $ 1300 wird aber nicht gelöscht und befindet sich ebenfalls noch im Speicher. Damit bekommt der folgende Befehl einen Sinn:

C („compare", vergleichen): Zugegeben, so schön die Verschiebemöglichkeit mit „T" auch ist, eine Schwachstelle darf nicht verschwiegen werden: Sprungadressen werden nicht mit umgerechnet, außerdem kommt es in einem Maschinenspracheprogramm oft genug vor, daß Speicheradressen, die irgendwann aufgerufen werden sollen, in ein Low- und ein High-

Byte zerlegt werden. Diese hätten in einem neuen, durch die Verschiebung zugewiesenen Speicherbereich eine ganz andere Bedeutung, da sie sich noch immer auf Adressen im Originalprogramm beziehen. Verschiedene Maschinensprache-Monitore des C64 bieten hier eine komfortable Funktion, bei der nur die Differenz zwischen neuem und altem Speicherbereichsanfang in die Parameter-

VERGLEICHT SPEICHERINHALTE: „C"

Folge zu „T" aufgenommen werden muß, der Computer berechnet dann automatisch die neuen Einsprungstellen und paßt sie an. Das kann der „T"-Befehl des Monitors im C128 leider nicht, die Nachbesserung muß immer noch von Hand vorgenommen werden. Zur Vereinfachung können Sie den „C"-Befehl benutzen, der beide Speicherbereiche (das Original und den an eine andere Stelle verschobenen) miteinander vergleicht. Die Adressen, in denen voneinander abweichende Speicherinhalte stehen, werden auf dem Bildschirm aufgelistet. So haben Sie schnell herausgefunden, wo die zu ändernden Byte stehen, um nach deren Anpassung auch im neuen Speicherbereich das Programm

STARTEN UND FÜLLEN: „G" UND „F"

zum Laufen zu bringen. Da sind wir auch schon beim nächsten Thema:

G („go", Start eines Programmes): Der exakte Befehl lautet beispielsweise G 1300 und ist eigentlich nichts anderes als der vielleicht besser bekannte SYS-Befehl im BASIC. Es wird zu der angegebenen Adresse gesprungen und ein dort stehendes Maschinenspracheprogramm abgearbeitet. Statt „G" können Sie auch den Buchstaben „J" (von „jump", springen) verwenden, der Effekt ist derselbe.

F („fill", füllen, beschreiben): Damit läßt sich ein angegebener Bereich mit einem bestimmten Byte beschreiben, beispielsweise eine Null, um einen vorher benutzten Speicher zu löschen. Der Gesamtbefehl sieht so aus:

F 1300 1400 00

Natürlich kann statt „00" jeder andere Byte-Wert verwendet werden. Geben Sie doch nur zum Spaß im Monitor folgendes ein:

F 0400 07FE 2A

und betrachten Sie sich den Effekt ...

SPEICHER-, LADE- UND DISKETTEN-BEFEHLE

Um bequemes Arbeiten mit dem „Monitor" sicherzustellen, ist selbstredend auch die Kommunikation mit Ihrer angeschlossenen Floppy-Station vorgesehen. Es wäre doch ziemlich umständlich, dazu eigens wieder ins BASIC 7.0 zurückzuschalten zu müssen. Die Komfortanweisung des BASIC 7.0 können Sie hier nicht verwenden, im Prinzip läuft es hier so ab, wie wir es aus der Zeit mit dem C64 kannten. Nehmen wir den SAVE-Befehl:

S „(Prg. Name)" 08 1300 1450

„08" bedeutet die Gerätenummer des Speichergerätes, (es kann auch „09" für ein zweites angeschlossenes Laufwerk oder„01" für die Datasette sein). Die beiden Hexzahlen dahinter sind bei der Eingabe bindend, da sie die Anfangs- und Endadresse des zu speichernden Bereiches bilden. Das Programm wird damit absolut gespeichert, das heißt, die Anfangsadresse wird auf der Diskette gleich mit eingetragen, um beim späteren Laden das Programm auch gleich wieder an diesen Bereich transferieren zu können. Die Anweisung lautet kurz und bündig:

L „(Prg.Name)" 08

Soll erst ab einer späteren Adresse innerhalb des festgelegten Speicherbereichs geladen werden, so geben Sie noch eine Adresse mit an,

L „(Prg.Name)" 08 1400

Wie schon in BASIC – oft gefordert, doch nie praktiziert, stimmt's? – läßt sich auch im Monitor gerade gespeichertes Programm (bzw. die abgespeicherten Bytewerte) „verifizieren", also überprüfen, ob sich Übertragungsfehler oder sonstige „Motten" eingeschlichen haben. Die Anweisung im Klartext:

V „(Prg.Name") 08

HAUPTPERSON: DER „KLAMMERAFFE"

Richtiger heißt dieses eigenartige Zeichen „At Sign" und signalisiert dem C128 im Monitor-Modus, daß Sie etwas mit der Floppy vorhaben. Wollen Sie nur das Directory einer Diskette betrachten, so geben Sie ein:

8, $

Die Anzeige des Floppy-Fehlerkanals erhalten Sie mit: „ " bekannt aus dem BASIC 7.0 „Print DS$". So bequeme Anweisungen wie RENAME, SCRATCH, HEADER können Sie hier nicht verwenden. Aber den ganzen Befehlsrattenschwanz zur Kommunikation mit einer Diskettenstation im Commodore-BASIC können Sie sich trotzdem sparen: Nie mehr OPEN 1,7,15 und CLOSE 1, das alles erledigt in dem Fall der Klammeraffe.

VERSCHIEDENE SYMBOLE – VERSCHIEDENE ZAHLENSYSTEME

Bei der Arbeit mit dem C128-Monitor brauchen Sie nicht unbedingt Hexadezimalzahlen anzugeben, die mit einem „$"-Symbol beginnen, es geht auch mit Dezimalzahlen. Sie müssen diese nur mit einem " + "-Zeichen (Plus) am Anfang versehen. Beispiel: Die Eingabe „M 1300 1350" bewirkt dasselbe wie „M + 4864+ 4944".

Es gibt noch zwei weitere Darstellungsarten von Zahlen, die oktalen (Symbol: &) und die binären (Symbol: %), in der Regel werden diese aber nicht benutzt.

WICHTIG: MASCHINENSPRACHE-KENNTNISSE

Sie haben bei unserem Streifzug sicher gesehen, welch schönes Programmierwerkzeug uns der C128 mit seinem Tedmon zur Verfügung stellt, allerdings sind hier Kenntnisse der Maschinensprache für die Mikroprozessoren 6510/8502, die praktisch identisch sind, Grundvoraussetzung. Nur dann können Sie diesen Monitor mit seinen Möglichkeiten so nutzen, daß Sie auch Spaß daran haben. *t.ch.* ■

C 64 GRAFIK AUF DEM 128?

Jede Menge Grafik- und „Mal"-Programme gibt's für den 64er, eines besser als das andere. Beim C 128 sieht das Software-Angebot bislang leider recht mager aus.

Nun ist es sicher kein Problem, solche Grafik-Tools im C-64-Modus des C 128 zu benutzen und damit Bilder und selbstgezeichnete Grafiken zu erstellen, diese abzuspeichern und in eigenen Programmen zu verwenden (Adventures, Dokumentationen mit Bildern unterlegt), aber bislang schien die Nutzung dieser „Kunstwerke" nur auf den C 64-Modus beschränkt.

BEISPIEL: HI-EDD 64

Doch das soll anders werden. Nehmen wir doch ruhig als Beispiel „Hi-Eddi", ein ausgezeichnetes Grafikprogramm für den C 64. Es arbeitet mit mehreren Bildschirmbereichen, die ab Adresse 8192 ($ 2000 hex.) beginnen, mehrere verschiedene Zeichenfarben sind darstellbar (wohlgemerkt, es ist ein Zeichenprogramm, das nur im HIRES-Modus, nicht aber mit der Multicolor-Darstellung arbeitet!). Diese farbigen Bilder benützen u.a. den Speicherbereich ab Adresse16384 ($ 4000 hex.).

Eines haben diese Grafikbilder aber mit einem im C 128-Modus erzeugten gemeinsam: sie beinhalten als C 64-Bild bit-, bzw. pixelweise dieselben Bildschirminformationen, wie sie auch der C 128 darstellt (kein Wunder, es ist ja auch derselbe Video-Chip!).

In der Normalkonfiguration liegen beim C 128 Grafiken (egal, ob hochauflösende oder Multicolor-Grafik) im Speicherbereich von 7168 ($ 1C00) bis 16383 ($ 3FFF). Wenn Sie nachrechnen, sind das genau 9216 Bytes, die zur Ablage der einzelnen Punkte der Bitmap plus der dazugehörigen Farbinformation gebraucht werden.

Wie nun ein mit einem 64er-Programm gezeichnetes Bild, das unter einem File-Namen auf Diskette gespeichert ist, in den C 128 bringen?

GRAFIK-BILDER WERDEN ABSOLUT GESPEICHERT UND GELADEN

Das ist gar nicht so schwer, kennt der große Bruder des C64 doch solch starke Befehle zum Speichern und Laden von genau begrenzten Speicherbereichen wie „BSAVE" und „BLOAD". Damit wird Bit für Bit und Byte für Byte als binäre Information innerhalb dieses Bereichs fortlaufend und in der richtigen Reihenfolge auf Diskette abgespeichert.

(Leider haben die Kassetten-Benutzer hier das Nachsehen). Unsere „Bitmap", die Grafiklandkarte des C 64 oder C 128, ist in diesem Falle eben so ein Speicherbereich. Allerdings: beim Abspeichern trägt der Computer in zwei Bytes auf der Diskette noch das

C-64 Grafiken im C 128

```
10 rem listing 1                    <oo>
20 rem lader c 64-grafik-           <bm>
30 rem bilder im s/w-modus          <ck>
40 rem belegte blocks auf           <po>
50 rem diskette < 33                <mj>
60 rem commodore welt/h.b.          <ak>
70 color0,7:color4,15:color1,2
:color5,2                           <fh>
80 scnclr:input"file-name";fl$      <mn>
90 graphic1,1                       <oc>
100 bload(fl$),on b0                <hi>
110 graphic0                        <kf>
120 scnclr:input"speichern (j/
n)";a$                              <dd>
130 ifa$="n"thenend                 <fk>
140 ifa$="j"thengraphic1:bsave
(fl$),onb0,p7168top16383:graph
ic0:end                             <fb>
```

```
10 rem listing 2                    <op>
20 rem verschiebt den basic-        <kh>
30 rem anfang nach $ 6400           <lj>
40 rem und laedt programm           <bg>
50 rem laut listing 3 nach !        <hg>
60 rem commodore welt/h.b.          <ak>
70 poke45,1:poke46,100:poke256
00,0                                <ho>
80 poke241,11:scnclr:printchr$
(19)"run"chr$(34)"main"chr$(34
)                                   <jh>
90 poke842,19:poke843,13:poke2
08,2:new                            <be>
```

Low- und High-Byte der Anfangsadresse des so gespeicherten Bildes (oder jedes anderen Speicherbereichs) auf der Diskette ein, um beim Wiederladen mit „BLOAD" (beim C 128) oder LOAD" (Name)",8,1 (beim C 64) gleich den richtigen Bereich Ihres Computers mit diesen Bytes zu belegen. Man nennt dies „absolutes Laden". (So werden auch Maschinenprogramme, die als binäres File auf Diskette vorliegen, abgespeichert und geladen!)

Da wir aber, um diese C 64-Grafikbilder zu laden, auch ein geeignetes Programm brauchen und dies zum besseren Verständnis ein Basic-Programm sein soll, gibt es Kompetenz-Schwierigkeiten mit der Stelle, ab der dieses Programm im Speicher stehen soll.

Der normale Basic-Anfang beim C 128 ist Adresse 7168, kommt aber nicht in Frage, da wir ja ein Grafikbild im hochauflösenden Modus laden möchten und zur Bildschirmdarstellung die „Bitmap" des C 128 brauchen (von 8168–16383). Der C 128 hat dies bereits in weiser Voraussicht geregelt und verlegt den Basic-Anfang nach 16384 ($ 4000), sobald er eine „GRAPHIC"-Anweisung empfängt, beispielsweise

```
10 rem listing 3                      <pa>
20 rem lader c 64-farbgrafik-         <ff>
30 rem bilder mit mehr als            <em>
40 rem 33 blocks auf diskette         <eo>
50 rem dieses programm bitte          <dn>
60 rem unter dem namen 'main'         <mb>
70 rem auf diskette abspeicher
n                                     <hf>
80 rem commodore welt/h.b.            <op>
90 color0,7:color4,15:color1,2
:color5,2                             <bj>
100 scnclr:input"file-name";fl
$                                     <go>
110 graphic1,1                        <cd>
120 bload(fl$),on b0                  <ji>
130 graphicclr:graphic1:bload(
fl$),onb0:graphic0                    <kn>
140 scnclr:input"speichern (j/
n)";a$                                <fa>
150 ifa$="n"thenend                   <kd>
160 ifa$="j"thengraphic1:bsave
(fl$),onb0,p7168top16383:graph
ic0:end                               <im>
```

```
10 rem laderoutine fuer               <jl>
20 rem c 128-grafikbilder             <af>
30 rem commodore welt/h.b.            <ho>
40 color0,7:color4,15:color1,2
:color5,2                             <nk>
50 ifpeek(4864)=1thengraphic1:
poke208,0:wait208,1:graphic0:i
nput"anderes bild (0=ja 1=nein
)";a:poke4864,a:graphic0:end          <jj>
60 scnclr:input"file-name";fl$        <mm>
70 graphic1,1                         <jn>
80 bload(fl$),on b0:poke4864,1        <ba>
90 poke208,0:wait208,1:graphic
0:end                                 <hd>
```

„GRAPHIC 1,1". Aber: Wir haben vorhin bereits besprochen, daß vor allem Farbgrafiken dieses C 64-Grafikprogramms eben genau an diesem Punkt beginnen, so daß unser dort stehendes Basic-Programm nur unliebsam stören würde.

BASIC-ANFANG VERLEGEN

Da bleibt uns gar nichts anders übrig, als den Basic-Speicheranfang des C 128 nach oben zu verschieben. Wohin und um wieviele Speicherstellen? Genau um die vorhin beschriebene Anzahl von 9216 Bytes. Rechnen wir nach: 16384 + 9216 = 25600. (in hexadezimaler Darstellung: $ 6400). Unsere Anweisung lautet also: POKE 45,1 : POKE 46,100: POKE 25600,0: NEW

Die Speicherstelle 45 beherbergt das Low-Byte (niederwertige Byte) der neuen Speicheradresse, die Adresse 46 das High-Byte (höherwertige Byte) in Dezimal-Darstellung. Die erste Speicherstelle des neuen Beginns des Basic-Speichers (25600) muß „0" betragen, daher wird dieser Wert in diese Zelle GePOKEt. Das folgende „NEW" initialisiert die neuen Werte.

Der Beginn der C 128-Bitmap samt Farbinformationen ist nun nicht mehr 7168, sondern 16384. So läßt sich nun das „Hi-Eddi"-Original-C 64-Bild ohne weiteres mit „BLOAD" in den C 128 laden, allerdings erst einmal mit den gültigen Farben. Nach Beendigung des Vorgangs müssen Sie mit der Anweisung „GRAPHIC CLR" dann das Bild erneut laden, diesmal geschieht das mit den Bytes der gültigen Zeichenfarbe (COLOR 1).

So erhält man durch „doppeltes" Laden des C 64-Originalbildes eine naturgetreue 128er-Kopie, die Sie jetzt mit den entsprechenden Anweisungen als C 128-Grafik abspeichern sollten:
BSAVE"128er-Grafik", ON BO, P 7168 TO P 16383

Einer weiteren Verwendung in eigenen 128er-Programmen steht damit nichts mehr im Wege.

Diese eben erwähnten Befehle und Anweisungen können Sie natürlich im Direktmodus eingeben, allerdings geht's bequemer mit unseren Listings für Ihren C 128.

ERLÄUTERUNG ZU DEN LISTINGS

Listing 1 ermöglicht Ihnen die Übernahme von sogenannten „Schwarz-Weiß"-Bildern solcher Grafikprogramme des C 64, hier ist es nicht nötig, den Basic-Anfang höher als ab Adresse 16384 zu setzen. Listing 2 und 3 sollten Sie für das Laden von Farbbildern einsetzen, (es funktioniert auch bei Multicolor-Bildern, getestet haben wir dies beim soeben erschienenen „Advanced OCP Art Studio" für den C 64!). Es wird zuerst der Basic-Anfang nach $ 6400 (25600) verlegt und das eigentliche Ladeprogramm der Grafiken automatisch nachgeladen. Nach Angabe eines Ihnen passenden File-Namens können Sie dann dieses Bild als C 128-Grafik auf Diskette abspeichern lassen.

Listing 4 zu guter Letzt bietet Ihnen die Möglichkeit, solche in den C 128 konvertierten Bilder (oder andere, reine C 128-Grafiken) wieder zu laden und sich anzusehen. Sollten Sie mit der Farbzusammenstellung nicht einverstanden sein, so ändern Sie bitte einfach die Werte der betreffenden COLOR-Anweisungen.

Wir hoffen, Ihnen mit diesem Beitrag als C 128-Benutzer eine Möglichkeit anhand gegeben zu haben, das Manko der

KEINE GARANTIE!

fehlenden Grafik- und Malprogramme für den C 128 (wir kennen bislang nur eines dafür, (Starpainter.128) auszugleichen. Nach dem Motto, „Mit dem C 64 malen, im C 128 verwenden!" Allerdings können wir keine Garantie übernehmen, ob es nach dieser Methode bei anderen Grafikprogrammen für den C 64 außer den beiden „Hi-Eddi"-Versionen und „Advanced OCP Art Studio" klappt, auch hier gilt der ewig junge Grundsatz:Probieren — und dann nochmals Probieren!

(hb)

FRACTALS: MATHEMATISCHE GRAFIK AUF DEM MONITOR

Wir kennen alle die bekannten geometrischen Formen wie Linie, Kreis, Rechteck, Kegel, Zylinder usw. usw. Alle diese Gebilde können mit mathematischen und physikalischen Formeln nach Größe, Umfang und Inhalt berechnet werden.

Wie sieht's aber in der Natur wirklich aus? Nach welcher Formel wollen Sie die Oberfläche eines Eichenblatts, den Umfang eines Berges oder den Rauminhalt eines Felsbrockens berechnen? Darüber streiten sich die Wissenschaftler, wir wollen uns da nicht einmischen, sondern einen Nutzen für uns und den Computer daraus ziehen. Die Erkenntnisse gipfeln darin, daß diesen erwähnten natürlichen Körpern oder Formen eine „gebrochene" Dimension zugeordnet wird, die sich in Zwischenbereichen zu den starr festgelegten, bislang bekannten mathematischen Gesetzen bewegt. Lassen Sie doch Ihren C 128 nach diesen neuen Erkenntnissen ein Bild auf dem Computer entwerfen, am besten im Multicolor-Modus!

BERECHNETE „UNORDNUNG" – RECHEN-MARATHON FÜR DEN COMPUTER!

Das folgende Listing definiert in Zeile 100 die Farbgebung, die Sie aber nach Ihrem persönlichen Geschmack anpassen können. Nach der Festlegung der Variablen wird in Zeile 150 die Anordnung der verschiedenen Zeichenfarb-Modi eingelesen (DATA-Zeile 340). Sie können die Anordnung beliebig verändern, nur nicht die Zahlen, die immer „1,2 oder 3" lauten müssen. Sie geben nämlich an, in welchen Zeichenfarb-Modus dann ein Punkt auf dem Bildschirm gesetzt wird, wenn gewisse Bedingungen erfüllt sind. Dies hängt im wesentlichen von den Variablen Z (Zeile 130) und E (Zeile 210) ab. Für das kontinuierliche Setzen der Zeichenpunkte sind die Zeilen 220, 230 und natürlich Zeile 250 zuständig. Eine Warnung gleich vorneweg: Für dieses Grafikbild muß der C 128 32000, zum Teil verschiedenfarbige, Bildpunkte (Pixels) auf der Multicolor-Bitmap setzen. Gehen wir davon aus, daß er für jeden Punkt nur eine Sekunde brauchen würde, so benötigt der C 128 zum Zeichnen dieses „Fractals" schon 9 Stunden! (Da Sie ja nicht unbedingt dabei sein müssen, können Sie das ruhigen Gewissens den Computer während der Nacht erledigen lassen!). Beschleunigen läßt sich das Ganze aber mit zwei in das Programm einzufügenden Zeilen:

90 FAST
265 SLOW

Mit dem „FAST"-Befehl veranlassen Sie Ihren

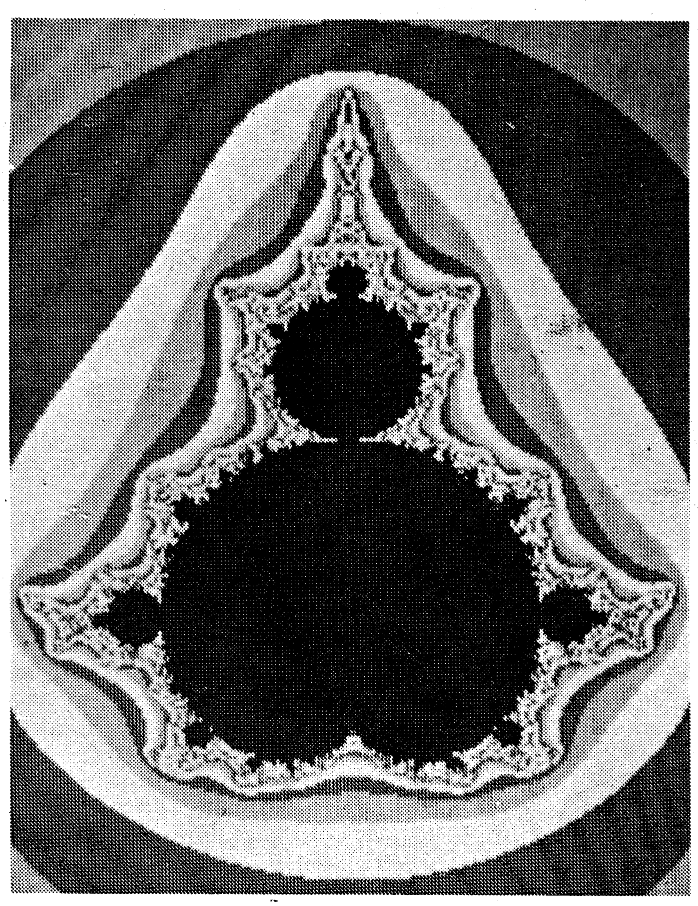
```
10 rem fractal ============128 <jm>
20 rem (c) commodore welt    == <al>
30 rem ======================= <ae>
40 rem      by h.b.          == <de>
50 rem                       == <if>
60 rem version 7.0 40z/ascii-- <bg>
70 rem 128 pc+datas./floppy == <hi>
80 rem                       == <he>
90 rem ======================= <aa>
110 color0,1:color4,1:color1,5
:color2,3:color3,11            <pk>
120 scnclr:graphic3,1          <ad>
130 a=20:b=23:d=1:dimc(a),x(b)
,y(b)                          <bh>
140 x1=d:y1=d:x2=-d:y2=-d:x3=0
.2:y3=0.5:z=100                <ck>
150 fx=(x1-x2)/159:fy=(y1-y2)/
199                            <eg>
160 fori=1toa:readc(i):next    <ec>
170 fori=0to199:forj=0to159    <ne>
180 x(0)=x2+j*fx:y(0)=y2+i*fy:
v=0                            <md>
190 x(v+1)=x(v)^2-y(v)^2-x3    <ea>
200 y(v+1)=x(v)*2*y(v)-y3      <kd>
210 v=v+1                      <cm>
220 e=x(v)^2+y(v)^2            <af>
230 ifv=a+1then270             <jc>
240 ife>zthen260               <kn>
250 goto190                    <fh>
260 drawc(v),j,i               <ie>
270 nextj,i                    <mp>
280 color4,15:poke208,0:wait20
8,1                            <jj>
290 graphic0:scnclr:print"flop
py eingeschaltet (j/n) ?"      <ia>
300 getkeyt$:ift$="n"then290   <jp>
310 ift$="j"then330            <bh>
320 goto300                    <lf>
330 graphic3:bsave"fractal",on
b0,p7168top16383               <bn>
340 end                        <ck>
350 data1,3,2,1,3,2,1,3,2,1,3,
2,1,3,2,1,3,2,1,3              <nj>
```

Fractal 2

```
10 rem fractal ============128 <jm>
20 rem (p) commodore welt    -- <hf>
30 rem ======================= <ae>
40 rem (c) hermann wellessen-- <ij>
50 rem      (nach einem      -- <in>
60 rem      amiga-programm)  == <hd>
70 rem version 7.0. 40z/ascii= <mp>
80 rem 128 pc+datas./floppy == <pa>
90 rem ======================= <aa>
100 color0,7:color4,7:color1,3
```

```
:color2,1:color3,2                    <lc>
110 scnclr:graphic3,1                 <hp>
120 xm=159:ym=95                      <jh>
130 readxc                            <cf>
140 readyc                            <hi>
150 readt                             <ln>
160 reads                             <oe>
170 readxl                            <go>
180 readyo                            <md>
190 readyu                            <bh>
200 readmk                            <ef>
210 ifmk=1thenrestore                 <jk>
220 dx=(xr-xl)/xm                     <ig>
230 dy=(yu-yo)/ym                     <km>
240 forn=0toym                        <hl>
250 y1=yo+n*dy                        <ge>
260 form=0toxm                        <kp>
270 x=xl+m*dx                         <id>
280 y=y1                              <ak>
290 k=0                               <nl>
300 x2=x*x                            <bn>
310 y2=y*y                            <lp>
320 y=2*x*y-yc                        <fo>
330 x=x2-y2-xc                        <dg>
340 k=k+1                             <ka>
350 if(k<t)and(x2+y2<s)then300        <kd>
360 u=m+10-n/2:ifu<0thenu=0           <ni>
370 v=n+80:ifv<0thenv=0               <am>
380 v1=v-3*(k-1):ifv1<0thenv1=
0                                     <hp>
390 draw3,u,vtou,v1                   <ln>
400 draw2,u+2,vtou+2,v1               <ob>
410 draw1,u,v1tou+2,v1                <ik>
420 nextm:nextn                       <hp>
430 z=0                               <hk>
440 gett$:z=z+1                       <fp>
450 ifz=10then110                     <gb>
460 ift$<>""then 110                  <nf>
470 goto440                           <ck>
480 data .82,-.5,30,30,-1.1,.3
3,-.72,.77,0                          <nk>
490 data.77,.6,22,40,-1.4,1.4,
-.9,1.2,0                             <pc>
500 data1.2,-.4,30,60,-1.15,1.
26,.47,.7,0                           <cn>
510 data1,.4,25,60,-.4,.26,-.4
7,.9,0                                <io>
520 data1,.5,25,60,-.77,.16,-.
37,.84,0                              <ej>
530 data1,0,20,60,-.15,.26,.47
,.9,1                                 <pn>
```

C 128, doppelt so schnell wie bisher zu arbeiten, die normale Taktfrequenz von 1 Hz wird auf 2 Hz erhöht. Einen Nachteil müssen Sie allerdings in Kauf nehmen: Der VIC-II-Chip kann dieser Geschwindigkeit nicht folgen und wird während der Zeit, in der der C 128 im FAST-Modus läuft, abgeschaltet. So können Sie dem Aufbau der „Fractal-Grafik" nicht zusehen, die Ausführungszeit wird aber auf gut die Hälfte verkürzt. Sind alle erforderlichen Grafikpunkte gesetzt, so wird in Zeile 265 mit „SLOW" wieder auf die sichtbare 40-Zeichendarstellung zurückgeschaltet.



Der C 128 wartet dann mit der fertigen Grafik auf Sie, die Sie per Tastendruck, nach der vorherigen Sicherheitsabfrage ob die Floppy eingeschaltet ist, auf Diskette abspeichern können.

Variieren Sie ruhig mit diesem Beispielprogramm, ändern Sie verschiedene Parameter (das geht ganz einfach, da diese in Form von numerischen Variablen in den Zeilen 120 und 130 abgelegt sind und sich somit automatisch dem Programm anpassen). Lediglich beim Ändern des Wertes der Variablen „A" müssen Sie darauf achten, daß auch mindestens soviele DATA-Werte in Zeile 340 angegeben sind. Außerdem sollten Sie vor jeder neuen Erzeugung eines „Fractals" den File-Namen in Zeile 330 umbenennen, z.B. durch Anhängen einer Zahl o.ä.

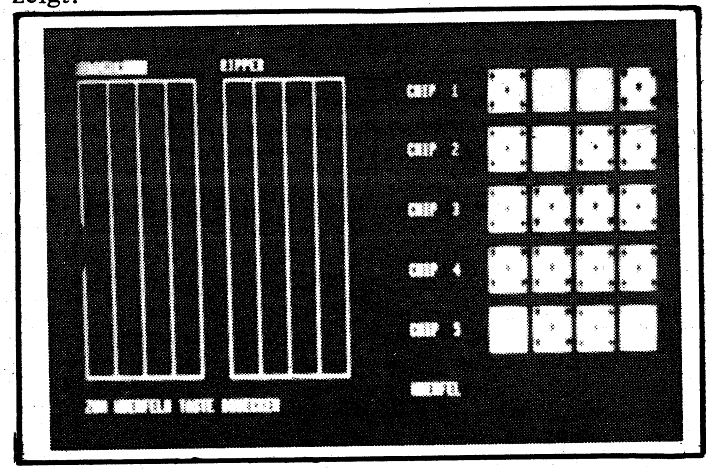
Falls Sie einmal Pech haben, und das Programm, nachdem der C 128 ca. 9 Stunden an der Grafik gearbeitet hat, geht durch irgendwelche Fehlbedienungen der Tastatur (oder der STOP/RESTORE-Taste) in den Direktmodus, so ist Ihr Bild noch lange nicht verloren.

Geben Sie ganz einfach die Listing-Zeile 320 (ohne Zeilennummer!) im Direktmodus ein, dann wird das fertige Grafikbild ebenso auf Diskette gespeichert (es belegt genau 37 Blöcke). *(hb)*

Puker

Dieses Spiel für den 128PC ist nur im 80-Zeichen-Modus lauffähig. Es ist ausschließlich für zwei Personen gedacht (von wegen Vereinsamung am Computer!). Obwohl die zu gewinnenden Chips wie Spielkarten aussehen, darf dieses Spiel nicht mit dem Glückspiel Poker verwechselt werden, damit hat es gar nichts zu tun.

Ihr Ziel sollte sein, so schnell wie möglich zehn Chips in Ihren Besitz zu bekommen.
In jeder Runde erhalten Sie einen Spielkartenwert, der in Ihrem Feld nach dem Würfeln ausgezeigt wird.
Haben Sie genügend Kartenwerte beisammen, können Sie mit dem Kauf der entsprechenden Chips beginnen. Diese werden auf der rechten Seite des Spielfeldes gezeigt.



Wenn jedes Symbol mindestens einmal in der für Sie zuständigen Spielspalte steht, kann gekauft werden, ansonsten meldet der Computer: Zu wenig Karten. Dabei haben Sie die Auswahl zwischen fünf Kombinationsmöglichkeiten.
Es kann also sein, daß die Wahl der Kombination 1 keinen Erfolg bringt, wohl aber die Reihung 3. Aufmerksames Beobachten des Bildschirms ist schon nötig.
Der Name des Spielers, der gerade an der Reihe ist, wird jeweils revers über dem Spielfeld angezeigt. Die Zahl dahinter gibt an, wieviele Chips Sie bereits besitzen.

BEDIENUNGSANLEITUNG

Nachdem Sie das Programm gestartet haben, müssen die Namen der beiden Spieler eingegeben werden. Auf Wunsch können Sie sich auch die Spielregeln noch einmal ansehen.
Dann wird das Spielfeld aufgebaut, auf der rechten Seite erscheint die Anordnung der Chips (insgesamt 4 x 5), die Sie „kaufen“ können. Allerdings erst dann, wenn Sie Sie im Besitz der entsprechenden Symbole sind. Das

Bitte lesen Sie weiter auf Seite 118

```
10 rem =======================128   <ob>
20 rem (p) commodore welt       ==   <hf>
30 rem ==========================   <mm>
40 rem (c) by                   ==   <pp>
50 rem     wolfgang werner      ==   <mg>
60 rem                          ==   <nd>
70 rem version 7.0   80z./ascii==   <gi>
80 rem pc-128 floppy/datasette ==   <ho>
90 rem ==========================   <km>

95 gosub 60000                       <jp>
100 vs=10                            <nj>
110 fast                             <nk>
120 scnclr                           <ni>
130 print"                    ppppp
u   u   k    k   eeeee   rrrrr
    "                                <fm>
140 print"                    p    p
u   u   k  k    e        r     r
    "                                <ce>
150 print"                    p    p
u   u   k k     e        r     r
    "                                <ad>
160 print"                    ppppp
u   u   kk      eeee     rrrrr
    "                                <pc>
170 print"                    p
u   u   k k     e        r  r
    "                                <gd>
180 print"                    p
u   u   k  k    e        r   r
    "                                <ne>
190 print"                    p
 uuu    k    k   eeeee   r     r
    "                                <ml>
200 print"

        "                            <ao>
210 print"

        "                            <fp>
220 print"

        "                            <kp>
230 print"

        "                            <pp>
240 print"                    ein spiel
 fuer den commodore pc 128
        "                            <kl>
250 print"

        "                            <ka>
260 print"                    geschrieb
en   von   wolfgang werner
       "                             <nc>
270 print"
         im haferfeld 6
        "                            <bp>
280 print"                "rn$"(c)
1987"rf$"
                          "          <lg>
290 print"                  "rn$"(p)
1987"rf$"            4708 kamen
                          "          <na>
300 print"

        "                            <de>
310 char,0,23,"...................
.  t   a   s   t   e   ..........
............"                        <kh>
320 getw$                            <fp>
330 sleep 1                          <jj>
340 ifw$<>""then 400                 <gi>
350 char,0,23,"...................
.  t   a   s   t   e   ..........
............",1                       <jk>
360 getw$                            <jo>
370 sleep 1                          <nk>
380 ifw$<>""then 400                 <kl>
390 goto 310                         <cd>
400 forx=10to23                      <gh>
410 char,0,x,"

                          "          <db>
420 next                             <mn>
430 char,0,12,"gespielt wird diese
s spiel mit 2 personen. "            <li>
440 char,0,14,"bitte`gebt nun die
namen ein :"                         <ld>
450 char,0,16,"spieler 1 :
              ":inputna$(1)          <hh>
460 char,0,18,"spieler 2 :
              ":inputna$(2)          <ni>
470 na$(1)=left$(na$(1)+"
   ",10)                             <he>
480 na$(2)=left$(na$(2)+"
   ",10)                             <ok>
490 char,0,21,"sind die regeln bek
annt ?  (j / n)"                     <ab>
500 getkeyw$                         <bk>
510 ifw$="n"thengosub2710            <gd>
520 :                                <eo>
530 a$(1)=rn$+"Z    Z"+c4$+left$(ql
$,5)+"  Z  "+c4$+left$(ql$,5)+"Z
 Z"+rf$                              <ja>
540 a$(2)=rn$+"S    S"+c4$+left$(ql
$,5)+"  S  "+c4$+left$(ql$,5)+"S
 S"+rf$                              <ej>
550 a$(3)=rn$+"A    A"+c4$+left$(ql
$,5)+"  A  "+c4$+left$(ql$,5)+"A
 A"+rf$                              <hf>
560 a$(4)=rn$+"X    X"+c4$+left$(ql
$,5)+"  X  "+c4$+left$(ql$,5)+"X
 X"+rf$                              <gl>
570 :                                <lc>
```

```
580 he(1)=0                                <bd>
590 pi(1)=0                                <ib>
600 kr(1)=0                                <go>
610 scnclr                                 <la>
620 char,0,0,na$(1),1                      <pf>
630 char,0,1,zf$+zv$+zv$+zv$+zh$+z
v$+zv$+zv$+zh$+zv$+zv$+zv$+zh$+zv$
+zv$+zv$+zd$                               <ia>
640 forc=2to20:char,0,c,"B"+s2$+s2
$+s2$+"B"+s2$+s2$+s2$+"B"+s2$+s2$+
s2$+"B"+s2$+s2$+s2$+"B":next               <fp>
650 char,0,21,zc$+zv$+zv$+zv$+zg$+
zv$+zv$+zv$+zg$+zv$+zv$+zv$+zg$+zv
$+zv$+zv$+zs$                              <gn>
660 char,20,0,na$(2)                       <nh>
670 char,20,1,zf$+zv$+zv$+zv$+zh$+
zv$+zv$+zv$+zh$+zv$+zv$+zv$+zh$+zv
$+zv$+zv$+zd$                              <md>
680 forc=2to20:char,20,c,"B"+s2$+s
2$+s2$+"B"+s2$+s2$+s2$+"B"+s2$+s2$
+s2$+"B"+s2$+s2$+s2$+"B":next              <ek>
690 char,20,21,zc$+zv$+zv$+zv$+zg$
+zv$+zv$+zv$+zg$+zv$+zv$+zv$+zg$+z
v$+zv$+zv$+zs$                             <dg>
700 if w$="w"thengosub3270                 <ad>
710 rem festlegung der chips               <cf>
720 gosub2120                              <an>
730 char,0,0,na$(1),1                      <no>
740 char,20,0,na$(2)                       <mn>
750 char,0,23,"zum wuerfeln taste
druecken":getkeyw$                         <fl>
760 char,0,23,"
          "                                <eb>
770 gosub2050                              <ge>
780 s1(a)=s1(a)+1:ifs1(a)>18thens1
(a)=18                                     <jg>
790 gosub3270                              <pm>
800 :                                      <ia>
810 :                                      <je>
820 :                                      <ki>
830 char,0,23,"
          "                                <jn>
840 char,0,22,"wollen sie einen ch
ip ?      "                                <na>
850 char,0,23,"(1 - 5 ) oder (0) f
uer keinen"                                <ne>
860 getkeyw$                               <lb>
870 char,0,22,"
          "                                <ce>
880 char,0,23,"
          "                                <da>
890 ifw$="0"then1320                       <ea>
900 ifw$="1"then960                        <hj>
910 ifw$="2"then1010                       <pc>
920 ifw$="3"then1060                       <jm>
930 ifw$="4"then1110                       <kp>
940 ifw$="5"then1160                       <fi>
950 goto 860                               <gb>
960 ifs1(1)<b(1)ors1(2)<b(2)ors1(3
)<b(3)ors1(4)<b(4)thenchar,0,23,"z
u wenig karten - taste":getkeyw$:c
har,0,23,"
   ":goto840                               <kl>
970 forc=1to4:s1(c)=s1(c)-b(c)             <ff>
980 next                                   <db>
990 s1=s1+1                                <ke>
1000 char,14,0:printusing"###";s1:
goto1210                                   <hd>
1010 ifs1(1)<c(1)ors1(2)<c(2)ors1(
3)<c(3)ors1(4)<c(4)thenchar,0,23,"
zu wenig karten - taste":getkeyw$:
char,0,23,"
    ":goto840                              <dk>
1020 forc=1to4:s1(c)=s1(c)-c(c)            <ih>
1030 next                                  <jg>
1040 s1=s1+1                               <od>
1050 char,14,0:printusing"###";s1:
goto1210                                   <dk>
1060 ifs1(1)<d(1)ors1(2)<d(2)ors1(
3)<d(3)ors1(4)<d(4)thenchar,0,23,"
zu wenig karten - taste":getkeyw$:
char,0,23,"
    ":goto840                              <hj>
1070 forc=1to4:s1(c)=s1(c)-d(c)            <id>
1080 next                                  <pk>
1090 s1=s1+1                               <bk>
1100 char,14,0:printusing"###";s1:
goto1210                                   <fp>
1110 ifs1(1)<e(1)ors1(2)<e(2)ors1(
3)<e(3)ors1(4)<e(4)thenchar,0,23,"
zu wenig karten - taste":getkeyw$:
char,0,23,"
    ":goto840                              <jp>
1120 forc=1to4:s1(c)=s1(c)-e(c)            <ek>
1130 next                                  <fo>
1140 s1=s1+1                               <pf>
1150 char,14,0:printusing"###";s1:
goto1210                                   <pd>
1160 ifs1(1)<f(1)ors1(2)<f(2)ors1(
3)<f(3)ors1(4)<f(4)thenchar,0,23,"
zu wenig karten - taste":getkeyw$:
char,0,23,"
    ":goto840                              <kl>
1170 forc=1to4:s1(c)=s1(c)-f(c)            <lp>
1180 next                                  <md>
1190 s1=s1+1                               <fp>
1200 char,14,0:printusing"###";s1:
goto1210                                   <ki>
1210 gosub2120                             <il>
1220 :gosub3270                            <dp>
1230 :                                     <np>
1240 :                                     <pd>
1250 :                                     <ah>
1260 :                                     <bl>
1270 :                                     <cp>
1280 :                                     <ee>
```

```
1290 :                                   <fi>
1300 ifs1>=vsthen3110                    <ga>
1310 goto 840                            <ph>
1320 a=int(rnd(0)*3+1):ifa=2then13
70                                       <ob>
1330 b=int(rnd(0)*4+1)                   <mn>
1340 vt=s1(b):s1(b)=s2(b):s2(b)=vt <og>
1350 char,0,23,"schikane : vertaus
chung":sleep1                            <cc>
1360 gosub2470                           <hc>
1370 char,0,0,na$(1)                     <go>
1380 char,20,0,na$(2),1                  <aa>
1390 char,0,23,"
     "                                   <da>
1400 char,0,23,"zum wuerfeln taste
 druecken":getkeyw$                      <pk>
1410 char,0,23,"
     "                                   <og>
1420 gosub2050                           <ph>
1430 s2(a)=s2(a)+1:ifs2(a)>18thens
2(a)=18                                  <nn>
1440 ifs2(1)>0thenforc=1tos2(1):ch
ar,22,c+2,"Z":next                       <gc>
1450 ifs2(2)>0thenforc=1tos2(2):ch
ar,26,c+2,"S":next                       <jb>
1460 ifs2(3)>0thenforc=1tos2(3):ch
ar,30,c+2,"A":next                       <in>
1470 ifs2(4)>0thenforc=1tos2(4):ch
ar,34,c+2,"X":next                       <fg>
1480 char,0,23,"
     "                                   <fe>
1490 char,0,22,"wollen sie einen c
hip ?  "                                 <md>
1500 char,0,23,"(1 - 5 ) oder (0)
fuer keinen"                             <ac>
1510 getkeyw$                            <hi>
1520 char,0,22,"
     "                                   <dh>
1530 char,0,23,"
     "                                   <bk>
1540 ifw$="0"then1970                    <cm>
1550 ifw$="1"then1610                    <he>
1560 ifw$="2"then1660                    <nm>
1570 ifw$="3"then1710                    <lp>
1580 ifw$="4"then1760                    <kd>
1590 ifw$="5"then1810                    <ai>
1600 goto 1510                           <ea>
1610 ifs2(1)<b(1)ors2(2)<b(2)ors2(
3)<b(3)ors2(4)<b(4)thenchar,0,23,"
zu wenig karten - taste":getkeyw$:
char,0,23,"
    ":goto1490                           <bi>
1620 forc=1to4:s2(c)=s2(c)-b(c)          <io>
1630 next                                <ek>
1640 s2=s2+1                             <cc>
1650 char,34,0:printusing"###";s2:
goto1860                                 <if>
1660 ifs2(1)<c(1)ors2(2)<c(2)ors2(
3)<c(3)ors2(4)<c(4)thenchar,0,23,"
zu wenig karten - taste":getkeyw$:
char,0,23,"
    ":goto1490                           <ei>
1670 forc=1to4:s2(c)=s2(c)-c(c)          <mn>
1680 next                                <kp>
1690 s2=s2+1                             <gg>
1700 char,34,0:printusing"###";s2:
goto1860                                 <ia>
1710 ifs2(1)<d(1)ors2(2)<d(2)ors2(
3)<d(3)ors2(4)<d(4)thenchar,0,23,"
zu wenig karten - taste":getkeyw$:
char,0,23,"
    ":goto1490                           <ic>
1720 forc=1to4:s2(c)=s2(c)-d(c)          <ji>
1730 next                                <bd>
1740 s2=s2+1                             <gp>
1750 char,34,0:printusing"###";s2:
goto1860                                 <bn>
1760 ifs2(1)<e(1)ors2(2)<e(2)ors2(
3)<e(3)ors2(4)<e(4)thenchar,0,23,"
zu wenig karten - taste":getkeyw$:
char,0,23,"
    ":goto1490                           <pk>
1770 forc=1to4:s2(c)=s2(c)-e(c)          <kl>
1780 next                                <hh>
1790 s2=s2+1                             <bp>
1800 char,34,0:printusing"###";s2:
goto1210                                 <ea>
1810 ifs2(1)<f(1)ors2(2)<f(2)ors2(
3)<f(3)ors2(4)<f(4)thenchar,0,23,"
zu wenig karten - taste":getkeyw$:
char,0,23,"
    ":goto1490                           <nl>
1820 forc=1to4:s2(c)=s2(c)-f(c)          <mo>
1830 next                                <nm>
1840 s2=s2+1                             <lj>
1850 char,34,0:printusing"###";s2:
goto1860                                 <be>
1860 gosub2120                           <bi>
1870 gosub3270                           <hc>
1880 :                                   <pi>
1890 :                                   <am>
1900 :                                   <ca>
1910 :                                   <de>
1920 :                                   <ej>
1930 :                                   <fn>
1940 :                                   <hb>
1950 ifs2>=vsthen3110                    <ij>
1960 goto 1490                           <id>
1970 a=int(rnd(0)*3+1):ifa=2then 2
020                                      <af>
1980 b=int(rnd(0)*4+1)                   <nl>
1990 vt=s1(b):s1(b)=s2(b):s2(b)=vt <nk>
2000 char,0,23,"schikane : vertaus
chung":sleep1                            <ba>
2010 gosub2470                           <af>
2020 char,0,23,"
```

```
     "                                  <cd>
2030 goto730                            <bc>
2040 end                                <hp>
2050 forx=1to10                         <ei>
2060 a=int(rnd(0)*4+1)                  <nj>
2070 a$=a$(a)                           <en>
2080 fory=1to250:next                   <dg>
2090 char,56,21,a$                      <ee>
2100 next                               <pk>
2110 return                             <bk>
2120 forc=1to4:b(c)=0:next:forc=1t
o4                                      <ic>
2130 a=int(rnd(0)*4+1):b$(c)=a$(a)
:b(a)=b(a)+1                            <ld>
2140 next:a=1                           <ap>
2150 forc=56to78step6                   <eh>
2160 char,c,1,b$(a):a=a+1               <da>
2170 next                               <ih>
2180 forc=1to4:c(c)=0:next:forc=1t
o4                                      <ei>
2190 a=int(rnd(0)*4+1):c$(c)=a$(a)
:c(a)=c(a)+1                            <ip>
2200 next:a=1                           <kh>
2210 forc=56to78step6                   <dk>
2220 char,c,5,c$(a):a=a+1               <mc>
2230 next                               <pp>
2240 forc=1to4:d(c)=0:next:forc=1t
o4                                      <nl>
2250 a=int(rnd(0)*4+1):d$(c)=a$(a)
:d(a)=d(a)+1                            <aa>
2260 next:a=1                           <ca>
2270 forc=56to78step6                   <oa>
2280 char,c,9,d$(a):a=a+1               <mn>
2290 next                               <hh>
2300 forc=1to4:e(c)=0:next:forc=1t
o4                                      <ge>
2310 a=int(rnd(0)*4+1):e$(c)=a$(a)
:e(a)=e(a)+1                            <nl>
2320 next:a=1                           <li>
2330 forc=56to78step6                   <bj>
2340 char,c,13,e$(a):a=a+1              <oe>
2350 next                               <pa>
2360 forc=1to4:f(c)=0:next:forc=1t
o4                                      <ho>
2370 a=int(rnd(0)*4+1):f$(c)=a$(a)
:f(a)=f(a)+1                            <np>
2380 next:a=1                           <cn>
2390 forc=56to78step6                   <ak>
2400 char,c,17,f$(a):a=a+1              <kp>
2410 next                               <gi>
2420 a=1                                <ib>
2430 forc=2to20step4:char,45,c,"ch
ip ":printa:a=a+1                       <ff>
2440 next                               <kf>
2450 char,45,22,"wuerfel"               <je>
2460 return                             <nj>
2470 rem vertauschen                    <ah>
2480 char,0,23,"getauscht wird   :
```

```
     ":sleep1                           <jm>
2490 ifb=1then print"Z"                 <dl>
2500 ifb=2then print"S"                 <lj>
2510 ifb=3then print"A"                 <jb>
2520 ifb=4then print"X"                 <nn>
2530 forc=1to1000:next                  <fm>
2540 forc=1to18:char,2,c+2," ":nex
t                                       <ca>
2550 forc=1to18:char,6,c+2," ":nex
t                                       <bl>
2560 forc=1to18:char,10,c+2," ":ne
xt                                      <ed>
2570 forc=1to18:char,14,c+2," ":ne
xt                                      <aa>
2580 ifs1(1)>0thenforc=1tos1(1):ch
ar,2,c+2,"Z":next                       <eg>
2590 ifs1(2)>0thenforc=1tos1(2):ch
ar,6,c+2,"S":next                       <mh>
2600 ifs1(3)>0thenforc=1tos1(3):ch
ar,10,c+2,"A":next                      <ne>
2610 ifs1(4)>0thenforc=1tos1(4):ch
ar,14,c+2,"X":next                      <lb>
2620 forc=1to18:char,22,c+2," ":ne
xt                                      <nm>
2630 forc=1to18:char,26,c+2," ":ne
xt                                      <aa>
2640 forc=1to18:char,30,c+2," ":ne
xt                                      <hb>
2650 forc=1to18:char,34,c+2," ":ne
xt                                      <ho>
2660 ifs2(1)>0thenforc=1tos2(1):ch
ar,22,c+2,"Z":next                      <cc>
2670 ifs2(2)>0thenforc=1tos2(2):ch
ar,26,c+2,"S":next                      <gb>
2680 ifs2(3)>0thenforc=1tos2(3):ch
ar,30,c+2,"A":next                      <nn>
2690 ifs2(4)>0thenforc=1tos2(4):ch
ar,34,c+2,"X":next                      <bj>
2700 return                             <ll>
2710 scnclr                             <cj>
2720 print"                        spi
elregeln fuer    "rn$"  p  u  k  e
  r  "rf$                               <je>
2730 print"                        "zv
$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$
zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$z
v$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv
$zv$zv$                                 <io>
2740 print"


"                                       <lo>
2750 print"                        zie
l dieses spieles ist es so schnell
"                                       <fk>
2760 print"                        wie
 moeglich 10 chipse in seinen be-
"                                       <dp>
2770 print"                        sit
```

```
z zu bringen.
"                                        <cp>
2780 print"

"                                        <pp>
2790 print"                          in
jeder runde bekommen sie eine     "  <oi>
2800 print"                          spi
elkarte. den besitz der spielkarte
n                                        <ca>
2810 print"                          koe
nnen sie jeweils in der linken ta-
"                                        <ml>
2820 print"                          bel
le erkennen. haben sie genug karte
n                                        <kh>
2830 print"                          koe
nnen sie beginnen chips zu kaufen.
"                                        <hg>
2840 print"

"                                        <of>
2850 print"                          die
 chips sehen sie am rechten rand.
"                                        <jj>
2860 print"

"                                        <ic>
2870 print"

"                                        <nc>
2880 print"

"                                        <cd>
2890 print"                          ...
........ t a s t e  .............
"                                        <ao>
2900 getkeyw$:scnclr                     <cd>
2910 print"                          spi
elregeln fuer   "rn$"  p  u  k  e
  r  "rf$                                <on>
2920 print"                          "zv
$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$
zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$z
v$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv
$zv$zv$                                  <ob>
2930 print"

"                                        <le>
2940 print"                          der
 name des aktuellen spielers ist
"                                        <pj>
2950 print"                          jew
eils revers ueber dem kartenfeld
"                                        <gl>
2960 print"                          ang
egeben. der besitz an chipse steht
"                                        <ff>
2970 print"                          jew
eils rechts neben dem namen.
"                                        <oo>
2980 print"

"                                        <ei>
2990 print"                          abe
r so einfach wie es sich bis hier-
"                                        <bb>
3000 print"                          her
anhoert ist es nun doch wieder
"                                        <pe>
3010 print"                          nic
ht. nach jeder runde ist eine
"                                        <ao>
3020 print"                          sch
ikane eingebaut. wie dies funktion
iert                                     <fl>
3030 print"                          wer
den sie schon merken.
"                                        <dp>
3040 print"

"                                        <cg>
3050 print"                          wen
n sie nun eine "rn$"  t  a  s  t
e  "rf$"  "                              <md>
3060 print"                          dru
ecken kann es losgehen.
"                                        <kn>
3070 print"

"                                        <be>
3080 print"                          ...
........ viel spass  ..........
"                                        <hk>
3090 getkeyw$:return                     <jj>
3100 :                                   <ik>
3110 rem gewinnwertung                   <jm>
3120 scnclr                              <ga>
3130 print"                 p u k e r
 :      g e w i n n a u s w e r t
u n g"                                   <co>
3140 print"            "zv$zv$zv$zv$
zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$z
v$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv
$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$zv$
zv$zv$zv$zv$zv$zv$;                      <hm>
3145 printzv$zv$zv$zv$zv$zv$zv$          <nf>
3150 printc4$c4$c4$"               gewo
nnen hat dies spiel :";" ";              <mm>
3160 ifs1>s2thenprintna$(1):elsepr
intna$(2)                                <pk>
3170 printc4$c4$c4$"               zwei
ter sieger ist      :";" ";              <ai>
3180 ifs1<s2thenprintna$(1):elsepr
intna$(2)                                <oe>
3190 printleft$(qd$,4)"               "
```

```
rn$" n "rf$"eues spiel"                <ha>
3200 printc4$"          "rn$" w "r
f$"eiter spielen bis ";vs+10           <dg>
3210 printc4$"          "rn$" e "r
f$"nde"                                <ph>
3220 getw$                             <fm>
3230 if w$="w"thenvs=vs+10:goto530     <kp>
3240 if w$="n"thenrun                  <gf>
3250 ifw$="e"thenscnclr:print"auf
wiedersehen !":new                     <dn>
3260 goto 3220                         <je>
3270 rem                               <de>
3280 forc=1to18:char,2,c+2," ":nex
t                                      <jh>
3290 forc=1to18:char,6,c+2," ":nex
t                                      <ie>
3300 forc=1to18:char,10,c+2," ":ne
xt                                     <in>
3310 forc=1to18:char,14,c+2," ":ne
xt                                     <en>
3320 ifs1(1)>0thenforc=1tos1(1):ch
ar,2,c+2,"Z":next                      <gg>
3330 ifs1(2)>0thenforc=1tos1(2):ch
ar,6,c+2,"S":next                      <bi>
3340 ifs1(3)>0thenforc=1tos1(3):ch
ar,10,c+2,"A":next                     <kd>
3350 ifs1(4)>0thenforc=1tos1(4):ch
ar,14,c+2,"X":next                     <in>
3360 forc=1to18:char,22,c+2," ":ne
xt                                     <jo>
3370 forc=1to18:char,26,c+2," ":ne
xt                                     <nc>
3380 forc=1to18:char,30,c+2," ":ne
xt                                     <co>
3390 forc=1to18:char,34,c+2," ":ne
xt                                     <el>
3400 ifs2(1)>0thenforc=1tos2(1):ch
ar,22,c+2,"Z":next                     <cl>
3410 ifs2(2)>0thenforc=1tos2(2):ch
ar,26,c+2,"S":next                     <kc>
3420 ifs2(3)>0thenforc=1tos2(3):ch
ar,30,c+2,"A":next                     <aa>
3430 ifs2(4)>0thenforc=1tos2(4):ch
ar,34,c+2,"X":next                     <nb>
3440 char,14,0:printusing"###";s1      <lm>
3450 char,34,0:printusing"###";s2      <hl>
3460 return                            <lb>
60000 rem nachspann ==============     <pj>
60010 rem *farbcodes/steuercodes *     <kg>
60020 c4$=chr$(017):rn$=chr$(018)      <pj>
60030 rf$=chr$(146):c1$=chr$(157)      <ep>
60040 rem ***zeichensatz/graphik *     <lm>
60050 s2$=chr$(160):zc$=chr$(173)      <cc>
60060 zd$=chr$(174):zf$=chr$(176)      <je>
60070 zg$=chr$(177):zh$=chr$(178)      <fh>
60080 zs$=chr$(189):zv$=chr$(192)      <ap>
60090 rem *********zeichenfolgen *     <cl>
60100 for q=1 to 40                    <jn>
60110 qd$=qd$+c4$:ql$=ql$+c1$          <ag>
60120 next q                           <fp>
60130 return                           <ch>
```

Puker

Fortsetzung von Seite 112

Konto der Symbole wird für jeden Mitspieler auf der linken Bildschirmhälfte geführt.
In jeder Runde erhalten Sie durch Würfeln (auf Tastendruck) ein Symbol Ihrem Konto gutgeschrieben. So sammeln Sie ein entsprechendes Guthaben.
Aber Vorsicht! Als „Schikane“ kann pro Runde Ihr Guthaben an einem bestimmten Symbol mit dem Ihres Gegners vertauscht werden (oder umgekehrt).
Darauf haben Sie keinen Einfluß, das legt der Zufallsgenerator fest. Wir können Ihnen daher nur empfehlen, sobald wie möglich einen Chip zu kaufen.

Dabei geschieht folgendes:

1. Das Punkte-(Chip-)Konto des jeweiligen Spielers wird erhöht. (Das ist die Zahl rechts neben dem Spielernamen).

2. Die verwendeten Symbole werden dem kaufenden Spieler abgezogen, er muß sie also durch Würfeln erneut erreichen.

3. Die Chip-Anordnung wird wieder neu festgelegt.

Wer als erster zehn Punkte gesammelt hat, ist der Gewinner. Nun werden Sie vom Computer gefragt, ob Sie weiterspielen möchten. Wenn Sie mit Ja antworten, bleibt der erreichte Punktestand erhalten. Das weitere Ziel ist jetzt die nächste Zehner-Grenze.

Wolfgang Werner/hb □

Börsen-Spiel

Sich den „scharfen“ Wind der Hochfinanz einmal um die Nase wehen lassen, die Wallstreet ins Wohnzimmer holen, das können Sie jetzt mit Ihrem Computer und diesem Programm.
Dieses grafisch gut gelungene Spiel für den C 128 läuft nur im 40-Zeichen-Modus und ist für mindestens zwei, höchstens aber sechs Spieler gedacht. Nach dem Start des Spiels können Sie sich nochmals die Spielanleitung zu Gemüte führen, wenn Sie “J“ eingeben.
Das Auswahlmenü bietet Ihnen folgende Optionen, die mit den entsprechenden Funktionstasten angewählt werden müssen:
F 1 Spielbrett wird auf dem Bildschirm angezeigt
F 2 Aktienkauf
F 3 Aktionskarten
F 4 Aktienverkauf
F 5 allgemeine Übersicht
F 6 Aktionskarte ausspielen
F 7 Nächster Spieler
F 8 Spiel beenden

SPIELVERLAUF

Dieses Spiel simuliert den Alltag an einer Aktienbörse. Durch geschickten Kauf und Verkauf von Aktien sollen Sie möglichst großen Gewinn erzielen. Mit der Hilfe von Aktionskarten können Sie direkt die Aktienkurse manipulieren! Aber aufgepaßt: Die anderen Mitspieler sind auch noch da!
Sind Sie an der Reihe, können Sie im Hauptmenü frei wählen, was Sie gerade tun möchten: Aktien kaufen und verkaufen, das Spielbrett oder Ihre Aktienübersicht anschauen. Allerdings sollten Sie zwei wichtige Regeln beachten:
- pro Spielrunde ist obligatorisch eine, aber nur eine Aktionskarte zu setzen
- vor dem Ausspielen der Aktionskarte gekaufte Aktien können nicht in der selben Runde verkauft werden. (Das gilt natürlich auch umgekehrt.)

DIVIDENDE UND KONKURS

Ein Aktienkurs übersteigt die 100-DM-Grenze. Die Differenz zwischen dem Aktienkurs und 100 wird nun in Form einer Dividende an alle betroffenen Aktionäre ausbezahlt. Der Aktienkurs fällt auf 100 DM zurück.
Fällt hingegen ein Aktienkurs unter die 10-DM-Grenze, bedeutet dies den Konkurs der betreffenden Firma. Alle Aktionäre verlieren diese Aktien. Pro Aktie muß zudem die Gebühr von 8 DM entrichtet werden. Wer kein bares Geld besitzt, muß andere Aktien verkaufen. Der Computer verkauft automatisch soviele Aktien, bis die Schuld getilgt ist. Hat jemand nach diesem Verkauf immer noch Schulden, scheidet er aus.

MELDUNGEN

Natürlich sind die Aktienkurse nicht alleine von Ihnen, also den Spielteilnehmern, abhängig. Nach jeder Spielrunde besteht die Möglichkeit einer sogenannten Meldung. Diese politische oder wirtschaftliche Mitteilung beeinflußt natürlich die Aktienkurse. Zu Beginn können Sie selber die Häufigkeit dieser Meldungen bestimmen.

SPIEL- UND BEDIENUNGSBESONDERHEITEN

- Um das Spiel zu erschweren ist es möglich, am Anfang einen Teil der Aktionskarten zugedeckt zu erhalten. Diese werden dann nach und nach (pro Runde eine) vom Computer aufgedeckt. Für Sie ist es jetzt schwieriger, Ihre Spieltaktik gleich am Anfang festzulegen.
- Sind Sie an der Reihe? Dann sollten Sie die anderen Mitspieler vom Bildschirm wegschicken, damit Sie Ihre Transaktionen alleine ausführen können. Auf diese Weise weiß niemand außer Ihnen, welche Aktien und welche Aktionskarten Sie gerade besitzen.
- Wollen Sie ein Untermenü frühzeitig verlassen, drücken Sie einfach Return. Diese Leereingabe führt Sie ins Hauptmenü zurück.
- Beim Menü „Aktienkauf“ kann bei der Frage nach der Anzahl mit “M“ geantwortet werden. Der Computer kauft für Sie nun die maximal mögliche Menge. Bei „Aktienverkauf“ werden auf die Antwort “A“ alle vorhandenen Aktien der betreffenden Firma verkauft.
- Keine Angst vor Fehleingaben und Regelverstößen! Diese werden vom Computer erkannt und verhindert.

(Stefan Schäli)

```
10 rem boersenspiel========128         <pc>
20 rem                        ==       <lh>
30 rem ======================          <ae>
40 rem (c) by                 ==       <pp>
50 rem            s. schaeli ==        <md>
60 rem                        ==       <ah>
70 rem version 7.0 40z/ascii==         <nn>
80 rem 128 pc + floppy        ==       <gk>
90 rem ======================          <aa>
100 trap4440:goto2540                  <ng>
110 rem f1 - spielbrett                <lk>
120 rem ***************                <eo>
130 fast:graphic1                      <da>
140 fori=1to5:spritei,1:nexti:slow     <ml>
150 color1,16:char,22,22,"space dr
uecken",1:getkeya$                     <ii>
160 fast:char,2,22,"         ":char,
11,22,"          ":graphic0:fori=1t
o5:spritei,0:nexti:slow:return         <nj>
170 rem f2 - aktienkauf                <bf>
180 rem ***************                <ob>
190 fast:window1,5,38,19:scnclr        <ih>
200 printro$c2$"aktienkauf
                "we$                   <ja>
210 print:printl1$;:fori=1to12:pri
ntl2$;:nexti:printl3$" "l1$;:fori=
1to21:printl2$;:nexti:printl3$         <nj>
220 printl4$tab(13)l5$" "l4$tab(37
)l5$                                   <ac>
230 printl6$;:fori=1to12:printl7$;
:nexti:printl8$" "l6$;:fori=1to21:
printl7$;:nexti:printl8$               <cc>
240 print:printl1$;:fori=1to36:pri
ntl2$;:nexti:printl3$                  <ca>
250 fori=1to5:printl4$tab(37)l5$:n
exti                                   <je>
260 printl6$;:fori=1to36:printl7$;
:nexti:printl8$                        <da>
270 fori=1to38:printl7$;:nexti         <bi>
280 window2,8,39,24:printc2$"firma
 :"tab(15)"anzahl :":gosub3490         <ok>
290 window2,12,39,24:print"vorhand
enes kapital   :  "dm$(sp)" dm"        <gp>
300 print:print"kaufbetrag
     :":fori=1to36:printun$;:nexti:
print:print"restkapital
:":slow                                <jh>
310 window10,8,13,9:fi$="":inputfi
$                                      <ij>
320 if fi$="" then return              <fg>
330 if fi$<>"a" and fi$<>"b" and f
i$<>"c" and fi$<>"d" and fi$<>"e"
then hi=4:gosub3570:printc2$:goto3
10                                     <hj>
340 gosub3410:an$="":window30,8,37
,9:inputan$:if an$="" then return      <dg>
350 an=val(an$)                        <nk>
360 if an$="m" then an=int(dm(sp)/
k(fi))                                 <ag>
370 if an<1 then return                <ig>
380 window27,14,39,24:kb=an*k(fi):
gosub3470:printkb$" dm"                <ae>
390 window27,16,39,24:rk=dm(sp)-kb
:gosub3470:printrk$" dm"               <ah>
400 if an*k(fi)>dm(sp) then hi=5:g
osub3570:goto190                       <mn>
410 if mk(sp,fi)=0 then hi=6:gosub
3570:hi=7:gosub3570:return             <mi>
420 hi=8:gosub3570:getkeya$:ifa$="
n" then goto190                        <md>
430 dm(sp)=dm(sp)-k(fi)*an:dm(sp)=
int(dm(sp)):ak(sp,fi)=ak(sp,fi)+an     <lf>
440 mv(sp,fi)=mv(sp,fi)-an:return      <nm>
450 rem f3 - aktionskarten             <bf>
460 rem ******************             <kf>
470 fast:window0,0,39,24:scnclr:nr
=0                                     <ak>
480 do:nr=nr+1                         <oc>
490 window pw(2*nr-1),pw(2*nr),39,
24:print:gosub3320                     <ga>
500 if len(ka$(sp,nr))=0 then goto
730                                    <ig>
510 if len(ka$(sp,nr))=1 then begi
n                                      <lp>
520 :o=fi:u=6:te$(1)="40 dm":te$(2
)=fi$                                  <eh>
530 :te$(3)="bel.  ":te$(4)="20 dm
"                                      <ph>
540 bend                               <in>
550 if len(ka$(sp,nr))=2 then begi
n                                      <jp>
560 :o=fi:u=6:te$(1)="2 x   ":te$(2
)=fi$                                  <ll>
570 :te$(3)="bel.  ":te$(4)="1/2
"                                      <mo>
580 bend                               <co>
590 if len(ka$(sp,nr))=3 then begi
n                                      <en>
600 :o=fi:u=6:te$(1)="50 dm":te$(2
)=fi$                                  <al>
610 :te$(3)="restl.":te$(4)="10 dm
"                                      <fh>
620 bend                               <mo>
630 if len(ka$(sp,nr))=4 then begi
n                                      <nj>
640 :o=6:u=fi:te$(1)="2 x   ":te$(2
)="bel.  "                             <jj>
650 :te$(3)=fi$:te$(4)="1/2   "        <if>
660 bend                               <gp>
670 if len(ka$(sp,nr))=5 then begi
n                                      <hd>
680 :o=6:u=fi:te$(1)="20 dm":te$(2
)="bel.  "                             <ih>
690 :te$(3)=fi$:te$(4)="30 dm"         <ig>
700 bend                               <ap>
```

```
710 if len(ka$(sp,nr))<4 then te$(
2)=te$(2)+"      ":else te$(3)=te$(
3)+"      "                         <ia>
720 gosub4220:if (kz(sp)>vz) and (
nr>2*kz-vz-kz(sp)) then printkg$:e
lse print ko$                       <kf>
730 loop until nr=kz                <bd>
740 slow:if f6<>1 then getkeya$     <mc>
750 return                          <gp>
760 rem f4 - aktienverkauf          <dd>
770 rem *****************           <gc>
780 fast:window1,5,38,19:scnclr     <na>
790 printro$c4$"aktienverkauf
                        "we$        <fe>
800 print:printl1$;:fori=1to12:pri
ntl2$;:nexti:printl3$" "l1$;:fori=
1to21:printl2$;:nexti:printl3$      <eb>
810 printl4$tab(13)l5$" "l4$tab(37
)l5$                                <bk>
820 printl6$;:fori=1to12:printl7$;
:nexti:printl8$" "l6$;:fori=1to21:
printl7$;:nexti:printl8$            <fb>
830 print:printl1$;:fori=1to36:pri
ntl2$;:nexti:printl3$               <pk>
840 fori=1to5:printl4$tab(37)l5$:n
exti                                <ck>
850 printl6$;:fori=1to36:printl7$;
:nexti:printl8$                     <lk>
860 fori=1to38:printl7$;:nexti      <gh>
870 window2,8,39,24:printc4$"firma
 :"tab(15)"anzahl :":gosub3470      <fp>
880 window2,12,39,24:print"vorhand
enes kapital   :  "dm$(sp)" dm"     <bf>
890 print:print"verkaufserloes
    :":fori=1to36:printun$;:nexti:
print:print"restkapital
:":slow                             <ma>
900 window10,8,13,9:fi$="":inputfi
$                                   <ib>
910 if fi$="" then return           <di>
920 if fi$<>"a" and fi$<>"b" and f
i$<>"c" and fi$<>"d" and fi$<>"e"
then hi=4:gosub3570:printc4$:goto9
00                                  <em>
930 gosub3410:an$="":window30,8,37
,9:inputan$:if an$="" then return   <fd>
940 an=val(an$)                     <dm>
950 ifan$="a"then an=ak(sp,fi)      <kn>
960 ifan<1 then return              <lp>
970 window27,14,39,24:vb=an*k(fi):
gosub3470:printvb$" dm"             <li>
980 window27,16,39,24:rk=dm(sp)+vb
:gosub3470:printrk$" dm"            <ck>
990 if an>ak(sp,fi) then hi=9:gosu
b3570:goto780                       <gc>
1000 if mv(sp,fi)<an then hi=10:go
sub3570:hi=11:gosub3570:return      <go>
1010 hi=8:gosub3570:getkeya$:ifa$=
"n" then goto780                    <da>
1020 dm(sp)=dm(sp)+k(fi)*an:dm(sp)
=int(dm(sp)):ak(sp,fi)=ak(sp,fi)-a
n                                   <pb>
1030 mk(sp,fi)=0:return             <dd>
1040 rem f5 - allgemeine uebersich
t                                   <di>
1050 rem *************************
*                                   <kn>
1060 fast:window1,5,38,19:scnclr    <ca>
1070 printro$c5$"allgemeine uebers
icht                "               <hn>
1080 print:print"firma"tab(16)"anz
ahl"tab(34)"wert"                   <ii>
1090 fori=1to38:printun$;:nexti     <kh>
1100 vb=0:kb=dm(sp)                 <eb>
1110 forfi=1to5:gosub3330:an=ak(sp
,fi):vb=an*k(fi):kb=kb+vb:gosub349
0                                   <nf>
1120 printtab(2)chr$(fa(fi))fi$tab
(12)an$tab(27)vb$" dm":nextfi       <eh>
1130 printc5$"vorhandenes kapital
:"tab(27)dm$(sp)" dm"               <bf>
1140 printtab(29)"CCCCCCCCC":print
"gesamtwert            :"tab(27)kb$"
 dm"                                <mk>
1150 printtab(29)"========="we$:fo
ri=1to38:printl7$;:nexti:slow       <ml>
1160 if di(sp)>0 then begin         <jo>
1170 :do                            <lj>
1180 :window1,21,38,23:printchr$(1
29)"achtung! dividenden der letzen
 runde:"                            <ch>
1190 :fi$=left$(di$(sp,(di(sp))),1
):gosub3410:print"firma "chr$(fa(f
i))fi$c5$" :"right$(di$(sp,(di(sp)
)),6)" dm ";                        <ig>
1200 :print"("mid$(di$(sp,(di(sp))
),2,3)" dm pro aktie)";:getkeya$    <kk>
1210 :di(sp)=di(sp)-1:loop until d
i(sp)=0                             <cf>
1220 bend                           <di>
1230 if ko(sp)>0 then begin         <bg>
1240 :do                            <nc>
1250 :window1,21,38,23:printc5$"ac
htung! konkurse der letzen runde:
 "                                  <fi>
1260 :fi$=left$(ko$(sp,(ko(sp))),1
):gosub3410:print"firma "chr$(fa(f
i))fi$c5$" : "right$(ko$(sp,(ko(sp
))),6)" dm ";:print"( 8 dm pro akt
ie)";:getkeya$                      <gc>
1270 :ko(sp)=ko(sp)-1:loop until k
o(sp)=0                             <cg>
1280 bend                           <cj>
1290 getkey a$:return               <bm>
1300 rem f6 - aktionskarte ausspie
len                                 <lp>
```

```
1310 rem *************************
***                                  <nf>
1320 if kz(sp)<ruthen hi=2:gosub35
70:return                            <dh>
1330 f6=1:gosub470:f6=0:rem alle a
ktionskarten zeigen                  <pg>
1340 nr$="":window1,24,39,24:scncl
r:printc9$"welche karte wollen sie
 setzen";:inputnr$                   <ll>
1350 if nr$="" then return           <od>
1360 nr=val(nr$)                     <fc>
1370 if nr=0 then return             <gc>
1380 scnclr:window0,0,39,24          <kh>
1390 if nr<0 or nr>9 then char,1,2
4,"diese nummer existiert nicht!":
sleep3:goto1340                      <pl>
1400 if ka$(sp,nr)="" then char,1,
24,"diese karte existiert nicht me
hr!":sleep3:goto1340                 <bk>
1410 if (kz(sp)>vz) and (nr>2*kz-v
z-kz(sp)) then char,1,24,"diese ka
rte ist noch zugedeckt!":sleep3:go
to1340                               <cj>
1420 fg$="":gosub3320:window1,23,3
9,24:print                           <of>
1430 if len(ka$(sp,nr))=1 then beg
in                                   <bp>
1440 :window1,24,39,24:scnclr:prin
t"firma deren kurs um 20 dm sinkt"
;:inputfg$:scnclr:window0,0,39,24    <gm>
1450 :gosub3610:if fe=1 then goto1
440                                  <ec>
1460 :k(fi)=k(fi)+40:fi$=fg$:gosub
3410:k(fi)=k(fi)-20                  <mk>
1470 bend                            <ce>
1480 if len(ka$(sp,nr))=2 then beg
in                                   <jj>
1490 :window1,24,39,24:scnclr:prin
t"firma deren kurs halbiert wird";
:inputfg$:scnclr:window0,0,39,24     <ff>
1500 :gosub3610:if fe=1 then goto1
480                                  <hc>
1510 :k(fi)=k(fi)*2:fi$=fg$:gosub3
410:k(fi)=(int(k(fi)/20))*10         <hm>
1520 bend                            <om>
1530 if len(ka$(sp,nr))=3 then beg
in                                   <mp>
1540 :k(fi)=k(fi)+60:fori=1to5:k(i
)=k(i)-10:nexti                      <ff>
1550 bend                            <gf>
1560 if len(ka$(sp,nr))=4 then beg
in                                   <pk>
1570 :window1,24,39,24:print"firma
 deren kurs verdoppelt wird";:inpu
tfg$:scnclr:window0,0,39,24          <cp>
1580 :gosub3610:if fe=1 then goto1
570                                  <hc>
1590 :k(fi)=(int(k(fi)/20))*10:fi$
=fg$:gosub3410:k(fi)=k(fi)*2         <dg>
1600 bend                            <co>
1610 if len(ka$(sp,nr))=5 then beg
in                                   <fn>
1620 :window1,24,39,24:scnclr:prin
t"firma deren kurs um 20 dm steigt
";:inputfg$:scnclr:window0,0,39,24   <nl>
1630 :gosub3610:if fe=1 then goto1
620                                  <hh>
1640 :k(fi)=k(fi)-30:fi$=fg$:gosub
3410:k(fi)=k(fi)+20                  <gj>
1650 bend                            <pg>
1660 ka$(sp,nr)="":kz(sp)=kz(sp)-1   <ch>
1670 fi=0:rem dividenden und konku
rs                                   <hj>
1680 fi=fi+1:gosub3330               <ll>
1690 if k(fi)>100 then begin         <jf>
1700 :di=k(fi)-100:k(fi)=110:gosub
3270:k(fi)=100                       <af>
1710 :fast:graphic1:color1,co(fi):
char,11,22,"dividende",1:gosub140:
i=0                                  <kf>
1720 :i=i+1:ifak(i,fi)<1 then goto
 1750                                <jg>
1730 ::di(i)=di(i)+1:di$(i,di(i))=
fi$:di$="  "+str$(di):di$(i,di(i))
=di$(i,di(i))+right$(di$,3):di$="
  "+str$(di*ak(i,fi))                <jb>
1740 ::di$(i,di(i))=di$(i,di(i))+r
ight$(di$,6):dm(i)=dm(i)+di*ak(i,f
i)                                   <fo>
1750 :if i<sz then goto1720          <hf>
1760 bend                            <la>
1770 if k(fi)<10 then begin          <kk>
1780 :k(fi)=0:gosub3270:k(fi)=50:f
ast:graphic1:color1,co(fi):char,2,
22,"konkurs",1:gosub140:i=0          <hp>
1790 :i=i+1:if ak(i,fi)<1 then got
o1840                                <hc>
1800 ::ko(i)=ko(i)+1:ko$(i,ko(i))=
fi$:ko$="     "+str$(8*ak(i,fi))     <lb>
1810 ::ko$(i,ko(i))=ko$(i,ko(i))+c
hr$(45)+right$(ko$,5):dm(i)=dm(i)-
8*ak(i,fi):ak(i,fi)=0                <mg>
1820 ::j=1:do while dm(i)<0:do whi
le (ak(i,j)>0) and (dm(i)<0):ak(i,
j)=ak(i,j)-1:dm(i)=dm(i)+k(j):loop
:j=j+1:if j=6 then exit              <jb>
1830 ::loop:if dm(i)<0 then au(i)=
2                                    <nf>
1840 :if i<sz then goto1790          <fm>
1850 bend                            <bk>
1860 gosub3270:if fi<5 then goto16
80:else:return                       <ce>
1870 rem f7 - naechster spieler      <ha>
1880 rem **********************      <ld>
1890 if kz(sp)>ru-1then hi=3:gosub
3570:goto2260:elsegoto2610           <hg>
```

```
1900 rem f8 - ganzes spiel beenden <df>
1910 rem ************************* <fl>
1920 window1,21,38,22:printlb$"wol
len sie das spiel wirklich schon":
print"beenden";:inputan$:if an$<>"
j"thengoto2260                     <jp>
1930 fast:ge(1)=0:forsp=1tosz:forf
i=1to5:dm(sp)=dm(sp)+ak(sp,fi)*k(f
i):nextfi                          <ip>
1940 fori=1tosz:ifdm(sp)>ge(i)ordm
(sp)=ge(i)then forj=sztoistep-1:ge
$(j+1)=ge$(j):ge(j+1)=ge(j):nextj:
ge$(i)=na$(sp):ge(i)=dm(sp):goto19
60                                 <jf>
1950 nexti                         <oe>
1960 nextsp                        <hm>
1970 gosub130:fast:f8=1:gosub2260:
window1,5,38,17:scnclr             <jh>
1980 printchr$(18)c8$"schlussueber
sicht                    ":print   <me>
1990 fori=1tosz:if ge(i)<=0 then g
oto2010                            <de>
2000 rk=ge(i):gosub3510:printchr$(
fa(i))"nr."i"  "ge$(i)tab(27)rk$"
dm"                                <hi>
2010 nexti:print:printchr$(154)"de
m sieger "ge$(1)" wird herzlich"   <hc>
2020 print"gratuliert!":slow:windo
w1,21,38,22:print"noch ein spiel";
:inputan$:if an$="j" then run      <ak>
2030 window1,21,38,22:scnclr:print
ro$"             auf wiedersehen!
        ":sleep4:window0,0,39,24   <if>
2040 scnclr:graphic1,1:graphic0:pr
intchr$(153):color4,14:color0,12:e
nd                                 <hd>
2050 rem meldungen                 <eh>
2060 rem *********                 <ij>
2070 m=int(rnd(1)*100+1):if m>hm t
hen return                         <bd>
2080 fast:window0,0,39,24:scnclr:p
rintlb$"achtung! achtung!":fori=1t
o17:printun$;:nexti                <ol>
2090 print:printc9$:fori=1to40:pri
ntl2$;:nexti:m=int(rnd(1)*12+1)    <nd>
2100 if m<5 then begin             <on>
2110 :kb=int(rnd(1)*m(2,m)):kb=(kb
+m(1,m))*10:gosub3500:kb$=right$(k
b$,7):fi=int(rnd(1)*5+1)           <ll>
2120 :gosub3330:printchr$(fa(fi))"
firma "fi$" meldet:":printm$(m)    <ba>
2130 :printro$"veraenderung des ak
tienkurses:"kb$" dm"               <mb>
2140 :printc9$;:fori=1to40:printl7
$;:nexti                           <oe>
2150 :k(fi)=k(fi)+kb:slow:getkeya$
:gosub1670                         <nc>
2160 bend                          <pg>
```

```
2170 if m>4 then begin             <cj>
2180 :printm$(m):print"resultieren
de kursschwankungen:":print:forfi=
1to5                               <hj>
2190 :kb=int(rnd(1)*m(2,m)):kb=(kb
+m(1,m))*10:gosub3500              <lb>
2200 :gosub3330:printchr$(fa(fi))r
o$"firma "fi$":
  "kb$" dm":k(fi)=k(fi)+kb:print:n
extfi                              <mj>
2210 printc9$:fori=1to40:printl7$;
:nexti:slow:getkeya$:gosub1670     <ih>
2220 bend                          <oh>
2230 return                        <al>
2240 rem hauptmenue                <hb>
2250 rem **********                <eh>
2260 fast:window0,0,39,24:scnclr:p
rintwe$"boersenspiel":fori=1to12:p
rintun$;:nexti                     <dm>
2270 window0,4,39,19               <hl>
2280 printl1$;:fori=1to38:printl2$
;:nexti:printl3$                   <ed>
2290 printl4$ro$lb$na$(sp);:fori=1
to(30-len(na$(sp))):print" ";:next
i:printtab(31)"runde:"str$(kz-ru+1
)we$rf$l5$                         <hb>
2300 printl4$tab(39)l5$            <ba>
2310 printl4$ro$c9$"f1  "c1$" spie
lbrett   "spc(2)c5$" allgemeine
"c9$"f5  "we$rf$l5$                <df>
2320 printl4$ro$c9$"    "c1$"
           "spc(2)c5$" uebersicht
"c9$"    "we$rf$l5$                <co>
2330 printl4$tab(39)l5$            <pp>
2340 printl4$ro$c9$"f2  "c2$" akti
enkauf   "spc(2)c6$" aktionskarte
"c9$"f6  "we$rf$l5$                <ng>
2350 printl4$ro$c9$"    "c2$"
           "spc(2)c6$" ausspielen
"c9$"    "we$rf$l5$                <ph>
2360 printl4$tab(39)l5$            <dd>
2370 printl4$ro$c9$"f3  "c3$" akti
ons-     "spc(2)c7$" naechster
"c9$"f7  "we$rf$l5$                <de>
2380 printl4$ro$c9$"    "c3$" kart
en       "spc(2)c7$" spieler
"c9$"    "we$rf$l5$                <kn>
2390 printl4$tab(39)l5$            <oc>
2400 printl4$ro$c9$"f4  "c4$" akti
en-      "spc(2)c8$" ganzes spiel
"c9$"f8  "we$rf$l5$                <kn>
2410 printl4$ro$c9$"    "c4$" verk
auf      "spc(2)c8$" beenden
"c9$"    "we$rf$l5$                <he>
2420 printl6$;:fori=1to38:printl7$
;:nexti:printl8$                   <pk>
2430 window0,20,39,24:printl1$;:fo
ri=1to38:printl2$;:nexti:printl3$  <gd>
```

```
2440 printl4$tab(39)l5$:printl4$ta
b(39)l5$                              <ja>
2450 printl6$;:fori=1to38:printl7$
;:nexti:printl8$                      <af>
2460 if f8=1 then return              <cf>
2470 slow:hi=1:gosub3570              <bj>
2480 getkeya$:a=val(a$):on a gosub
 130,190,470,780,1060,1320            <ld>
2490 ifa<7thengoto2260                <km>
2500 on (a-6) goto 1890,1920          <lj>

2510 goto2260                         <hi>
2520 rem hauptprogramm                <ja>
2530 rem *************                <lh>
2540 gosub3660:rem initialisierung
 und titelbild                        <on>
2550 gosub2720:rem anleitung und s
tart                                  <bm>
2560 do while ru>0:sp=0               <dl>
2570 :do                              <kp>
2580 ::sp=sp+1                        <hj>
2590 ::ifau(sp)=1orau(sp)=2thensp=
sp+1:ifsp>sz then goto2610            <pi>
2600 ::goto2260:rem hauptmenue        <gm>
2610 :loop until sp>=sz               <gg>
2620 :gosub2070:rem meldung           <ko>
2630 :sp=0                            <be>
2640 :sp=sp+1                         <in>
2650 :ifau(sp)=2thenslow:window0,0
,39,24:scnclr:printwe$"spieler "na
$(sp)" scheidet":print"wegen geldm
angels aus!!!":getkeya$:au(sp)=1      <jn>
2660 :forfi=1to5:mk(sp,fi)=1:mv(sp
,fi)=ak(sp,fi):nextfi:if sp<sz the
n goto2640                            <fe>
2670 :ru=ru-1                         <of>
2680 loop                             <pb>
2690 goto1930:rem ende                <pl>
2700 rem anleitung und start          <pf>
2710 rem ******************           <kn>
2720 window0,0,39,24:slow:scnclr:p
rintwe$"boersenspiel":fori=1to12:p
rintun$;:nexti                        <pb>
2730 print:input"wuenschen sie ein
e anleitung";an$:ifan$="j"thenbegi
n                                     <pk>
2740 fast:scnclr:print"anleitung":
fori=1to9:printun$;:nexti             <ce>
2750 print:print"dieses spiel simu
liert den alltag an":print"einer a
ktienboerse. bis max. 6 spieler"      <pm>
2760 print"koennen durch den kauf
und verkauf":print"der 5 moegliche
n aktiensorten das ge-":print"sche
hen beeinflussen. mit hilfe von"      <kb>
2770 print"aktionskarten kann jede
r mitspieler":print"direkt die akt
ienkurse manipulieren.":print"sieg
er des spiels wird, wer nach einer
"                                     <de>
2780 print"gewissen rundenzahl das
 groesste":print"vermoegen erwirts
chaftet hat.hier die":print"erklae
rung der wichtigsten menuepunkte:"    <nc>
2790 print:printc9$ro$"f1"c1$spc(1
)"spielbrett"rf$"auskunft ueber di
e neuesten":print"   aktienkurse d
er firmen a,b,c,d und e."             <dd>
2800 printc9$ro$"f2"c2$spc(1)"akti
enkauf"rf$" man erzielt gewinn dur
ch":print"   den kauf von preiswer
ten aktien vor"                       <md>
2810 print"   dem einsetzen der ak
tionskarte und":print"   durch den
 teurern derselben in der":print"
  naechsten runde."                   <bn>
2820 printc9$ro$"f3"chr$(150)spc(1
)"aktionskarten"rf$" uebersicht ue
ber den":print"   momentanen besta
nd aller karten, die"                 <kg>
2830 print"   fuer die kursmanipul
ation gebraucht":print"   werden k
oennen.";:slow:getkeya$:fast          <am>
2840 scnclr:printc9$ro$"f4"c4$spc(
1)"aktienverkauf"rf$" siehe aktien
kauf!                                 <od>
2850 print:printc9$ro$"f5"c5$spc(1
)"allgemeine uebersicht"rf$" momen
taner ge-":print"   samtwert aller
 aktien, sowie kapital-"              <hc>
2860 print"   bestand. ausserdem d
ie den spieler":print"   betreffen
den letzten dividenden- und":print
"   konkursmeldungen."                <pk>
2870 printc9$ro$"f6"c6$spc(1)"akti
onskarte ausspielen"rf$" je nach k
arte"                                 <ma>
2880 print"   gibt es verschiedene
 moeglichkeiten,":print"   die akt
ienkurse zu beeinflussen:"            <ol>
2890 print"   "we$chr$(206)chr$(20
5)" "l5$l4$c6$" der kurs steigt od
er sinkt um"                          <ih>
2900 print"   "we$l5$l4$" "chr$(10
9)chr$(110)c6$" den angegebenen be
trag."                                <ac>
2910 print:printwe$"   bel.  "c6$"
der zu beeinflussende kurs":print"
        kann ausgewaehlt werden."     <eh>
2920 printwe$"   restl."c6$"alle u
ebrigen aktienkurse":printspc(9)"s
inken um einen bestimmten":printsp
c(9)"betrag."                         <da>
2930 printwe$"   2x 1/2"c6$"der ak
tienkurs wird verdoppelt":printspc
(9)"oder halbiert."                   <ef>
```

```
2940 printc9$ro$"f7"c7$spc( 1)"naec
hster spieler"rf$" falls die aktio
ns-":print"   karte dieser runde a
usgespielt ist,"                     <ak>
2950 print"   kommt der naechste s
pieler an die":print"   reihe.":sl
ow:getkeya$:fast                     <dk>
2960 scnclr:printc9$ro$"f8"c8$spc(
1)"ganzes spiel beenden"rf$" auch
wenn nicht":print"   alle vorgeseh
enen aktionskarten aus-"             <ld>
2970 print"   gespielt sind, wird
das spiel abge-":print"   brochen.
 alle aktiendepots werden zum":pri
nt"   laufenden kurs bewertet und
der"                                 <bc>
2980 print"   sieger wird ermittel
t."                                  <dh>
2990 print:printlb$"drei wichtige
vorkommnisse:":print:print"1. stei
gt ein aktienkurs ueber 100 dm":pr
int"   so wird die differenz zwisc
hen dem"                             <ed>
3000 print"   kurswert und 100 dm
als "ro$"dividende":print"   ausbe
zahlt. der kurs faellt auf 100":pr
int"   dm zurueck."                  <og>
3010 print"2. faellt hingegen ein
kurs unter 10 dm,":print"   verlie
ren bei diesem "ro$"konkurs"rf$" d
er":print"   firma alle aktionaere
 die betreff-"                       <na>
3020 print"   enden aktien. sie be
zahlen zudem pro":print"   aktie 8
 dm. wer auch nach dem auto-"        <no>
3030 print"   matischen verkauf an
derer aktien":print"   durch den c
omputer die schuld nicht":print"
 begleichen kann, scheidet aus."     <ai>
3040 print"3. nach jeder runde (al
le spieler je 1x)":print"   werden
 die aktienkurse eventuell":print"
   durch eine meldung beeinflusst.
";:slow:getkeya$                     <oi>
3050 fast:scnclr:print"zum schluss
 noch eine einschraenkung:"          <ld>
3060 print:print"vor dem ausspiele
n "ro$"ge"rf$"kaufte aktien":print
"koennen in der gleichen runde nic
ht":printro$"ver"rf$"kauft werden
(und umgekehrt)."                    <fo>
3070 print:printc1$"viel glueck !!
!":slow:getkeya$:goto2700            <mg>
3080 bend                            <gf>
3090 window0,5,39,7:scnclr:print"w
ieviele spieler wollen am spiel te
il-":input"nehmen (2-6) ";sz         <ob>
3100 if sz<2 or sz>6 then goto3090   <jj>
3110 forsp=1tosz                     <nf>
3120 window0,8,39,9:scnclr:print"n
ame des"sp". mitspielers";:input n
a$(sp)                               <bn>
3130 if len(na$(sp))<1 or len(na$(
sp))>10 then goto3120:else nextsp    <lp>
3140 window0,10,39,12:scnclr:print
"wieviele aktionskarten (=runden)
soll":input"jeder bekommen (1-9) "
;kz                                  <pi>
3150 if kz<2 or kz>9 then goto3140   <dc>
3160 window0,13,39,16:scnclr:print
"wieviele davon sollen am anfang z
uge-":print"deckt sein (0-"right$(
str$(kz-1),1)") ";:inputvz           <mh>
3170 if vz<0 or vz>kz-1 then goto3
160                                  <dd>
3180 window0,17,39,19:scnclr:print
"wie gross soll die haeufigkeit vo
n":input"meldungen sein (0-100%) "
;hm                                  <fp>
3190 if hm<0 or hm>100 then goto31
80                                   <hd>
3200 fast:ru=kz:forsp=1tosz:kz(sp)
=kz:dm(sp)=300:fornr=1tokz:fi=int(
rnd(1)*5+1)                          <md>
3210 kt=int(rnd(1)*5+1):ka$(sp,nr)
=right$(str$(fi),1)+"     "          <gd>
3220 ka$(sp,nr)=left$(ka$(sp,nr),k
t):nextnr:forfi=1to5:mk(sp,fi)=1:m
v(sp,fi)=0:nextfi:nextsp:slow:retu
rn                                   <lo>
3230 rem hilfsprogramme              <cj>
3240 rem **************              <nl>
3250 rem kurse einzeichnen           <od>
3260 rem ****************            <ai>
3270 px=pk(24*(fi-1)+2*(int(k(fi)/
10+1))-1)                            <hk>
3280 py=pk(24*(fi-1)+2*(int(k(fi)/
10+1)))                              <ip>
3290 movsprfi,px,py:return           <ak>
3300 rem aus nummer wird firmenbuc
hstaben                              <id>
3310 rem *************************
*******                              <ie>
3320 fi=val(left$(ka$(sp,nr),1))     <mp>
3330 if fi=1 then fi$="a"            <bb>
3340 if fi=2 then fi$="b"            <nn>
3350 if fi=3 then fi$="c"            <jp>
3360 if fi=4 then fi$="d"            <fe>
3370 if fi=5 then fi$="e"            <cd>
3380 return                          <ba>
3390 rem aus buchstabe wird zahl     <ok>
3400 rem ***********************     <mg>
3410 if fi$="a" then fi=1            <jo>
3420 if fi$="b" then fi=2            <cf>
3430 if fi$="c" then fi=3            <jm>
3440 if fi$="d" then fi=4            <ci>
```

```
3450 if fi$="e" then fi=5                    <hi>
3460 return                                  <lb>
3470 rem darstellung von betraegen <cf>
3480 rem ************************* <kh>
3490 dm$(sp)=str$(dm(sp)):dm$(sp)=
"        "+dm$(sp):dm$(sp)=right
$(dm$(sp),8)                                 <pm>
3500 kb$=str$(kb):kb$="          "
+kb$:kb$=right$(kb$,8):rem kaufbet
rag/wert                                     <fg>
3510 rk$=str$(rk):rk$="          "
+rk$:rk$=right$(rk$,8):rem restkap
ital/gesamtkapital                           <dl>
3520 vb$=str$(vb):vb$="          "
+vb$:vb$=right$(vb$,8):rem vorhand
ener betrag/aktienwert                       <lp>
3530 an$=str$(an):an$="          "
+an$:an$=right$(an$,8):rem anzahl            <cn>
3540 return                                  <fb>
3550 rem hilfen                              <gd>
3560 rem ******                              <nk>
3570 window1,21,38,22:printlb$hi$(
hi)                                          <bj>
3580 sleep 2:scnclr:return                   <pk>
3590 rem kontrolle                           <ak>
3600 rem *********                           <mb>
3610 fe=0:if fg$=fi$ then char,1,2
4,"firmenkurs durch karte veraende
rt!":sleep3:fe=1                             <ck>
3620 if fg$<>"a" and fg$<>"b" and
fg$<>"c" and fg$<>"d" and fg$<>"e"
 then char,1,24,hi$(4):sleep3:fe=1 <hk>
3630 return                                  <ag>
3640 rem initialisierung und titel <da>
3650 rem ************************* <hf>
3660 fast:scnclr                             <ig>
3670 ro$=chr$(18):rf$=chr$(146):l1
$=chr$(111):l2$=chr$(183):l3$=chr$
(112):l4$=chr$(180):l5$=chr$(170):
l6$=chr$(108):l7$=chr$(175):l8$=ch
r$(186):un$=chr$(163)                        <hn>
3680 c1$=chr$(158):c2$=chr$(159):c
3$=chr$(28):c4$=chr$(156):c5$=chr$
(129):c6$=chr$(149):c7$=chr$(30):c
8$=chr$(31):c9$=chr$(155):we$=chr$
(5):lb$=chr$(154)                            <jn>
3690 key1,"1":key2,"2":key3,"3":ke
y4,"4":key5,"5":key6,"6":key7,"7":
key8,"8"                                     <di>
3700 dim z$(50),po(12),pk(120),pw(
18),ka$(6,9),hi$(12),f(50),m$(12),
m(2,12)                                      <jl>
3710 rem farben der firmen                   <fh>
3720 rem *****************                   <bo>
3730 co(1)=8:co(2)=6:co(3)=3:co(4)
=5:co(5)=15:rem color                        <lj>
3740 fa(1)=158:fa(2)=30:fa(3)=28:f
a(4)=156:fa(5)=154:fa(6)=155:rem c
hr$-farben                                   <io>
3750 fori=1to5:k(i)=50:nexti                 <al>
3760 rem verschiedene hilfen                 <dj>
3770 rem *******************                 <jo>
3780 hi$(1)="gewuenschtes bitte au
swaehlen!"                                   <cg>
3790 hi$(2)="karte dieser runde sc
hon eingesetzt!"                             <cg>
3800 hi$(3)="zuerst ihre aktionska
rte setzen!"                                 <jc>
3810 hi$(4)="diese firma existiert
 nicht!"                                     <dm>
3820 hi$(5)="sie haben zuwenig kap
ital!"                                       <kd>
3830 hi$(6)="diese sorte wurde ebe
n verkauft!":me$(7)="kauf erst nae
chste runde moeglich!"                       <ai>
3840 hi$(8)="alles ok?"                      <am>
3850 hi$(9)="es sind zuwenig aktie
n vorhanden!"                                <fj>
3860 hi$(10)="kein kauf und verkau
f von gleichen":hi$(11)="aktien in
 derselben runde moeglich!"                  <dp>
3870 gosub3950:rem titelbild                 <mn>
3880 gosub4050:rem spielfeld                 <od>
3890 forfi=1to5:gosub3270:nextfi:r
em kurse einzeichnen                         <de>
3900 gosub4370:rem kartendarstellu
ng                                           <pn>
3910 gosub4410:rem meldungen                 <nh>
3920 return                                  <em>
3930 rem titelbild                           <am>
3940 rem *********                           <bd>
3950 fast:graphic1,1:circle,6,6,6:
paint,6,6                                    <id>
3960 sshape ch$,0,0,23,20:graphic0
:color1,2:color0,1:color4,1:graphi
c1,1                                         <hd>
3970 restore4480:fori=1to20:readkx
,kf:color1,kf:gshapech$,kx,0:nexti <ca>
3980 restore4500:fori=1to10:readky
,kf:color1,kf:gshapech$,0,ky:nexti
:fori=1to10:readky,kf:color1,kf:gs
hapech$,304,ky:nexti                         <nh>
3990 restore4490:fori=1to20:readkx
,kf:color1,kf:gshapech$,kx,176:nex
ti                                           <jn>
4000 restore4520:fori=1to50:readz$
(i):nexti:restore4530:fori=1to48:r
eadf(i):ifi/12=int(i/12)then resto
re4530                                       <ik>
4010 nexti:readf(49),f(50):slow              <bf>
4020 restore4540:fori=1to50:readx,
y:color1,f(i):char,x,y,z$(i):sound
1,18000,1:sound1,7000,1:forj=1to70
:nextj:nexti:getkeya$:fast:return  <cf>
4030 rem spielfeld                           <kl>
4040 rem *********                           <ge>
```

```
4050 graphic1,1:fori=1to5:sprsavch
$,i:spritei,0,co(i),0,0,0,0:movspr
 i,0,0:nexti                          <gi>
4060 draw,17,20to300,20to300,150to
17,150to17,20                          <pe>
4070 draw,17,35to300,35:draw,17,13
5to300,135:draw,17,40to300,40:box,
2,160,300,199                          <fc>
4080 color1,2:box,12,170,75,187:bo
x,85,170,162,187:box,172,170,290,1
87                                     <pf>
4090 color1,8:paint,40,25:paint,40
,145                                   <na>
4100 color1,2:restore4580:fori=1to
12:read z$(i):read po(i)               <od>
4110 char,po(i),3,z$(i):char,po(i)
,17,z$(i):nexti                        <ke>
4120 fori=1to11:read po(i):draw,po
(i),40topo(i),135:nexti                <pn>
4130 fori=1to5 :read po(i):draw,17
,po(i)to300,po(i):nexti                <md>
4140 char,2,1,"boersenspiel"           <pn>
4150 color1,8:char,0,6,"a":color1,
6:char,0,8,"b":color1,3:char,0,11,
"c":color1,5:char,0,13,"d":color1,
15:char,0,15,"e"                       <pb>
4160 rem positionen der kurschips      <gk>
4170 rem ************************      <ed>
4180 restore4610:fori=1to120:read
pk(i):nexti:graphic0:return            <ej>
4190 rem kartenmaske                   <bf>
4200 rem ***********                   <pn>
4210 rem vorderseiten                  <dg>
4220 k1$=we$+l1$:fori=1to10:k1$=k1
$+l2$:nexti:k1$=k1$+l3$                <gd>
4230 k2$=l4$+" "+chr$(fa(o))+chr$(
206)+chr$(205)+" "+te$(1)+" "+we$+
l5$                                    <en>
4240 k3$=l4$+" "+chr$(fa(o))+l5$+l
4$+" "+te$(2)+we$+l5$                  <ee>
4250 k4$=l4$+"----------"+l5$          <ed>
4260 k5$=l4$+" "+chr$(fa(u))+l5$+l
4$+" "+te$(3)+we$+l5$                  <bo>
4270 k6$=c9$+right$(str$(nr),1)+"
"+chr$(fa(u))+chr$(109)+chr$(110)+
" "+te$(4)+" "+we$+l5$                 <cf>
4280 k7$=l6$:fori=1to10:k7$=k7$+l7
$:nexti:k7$=k7$+l8$                    <cc>
4290 k8$=chr$(17):fori=1to12:k8$=k
8$+chr$(157):nexti                     <ec>
4300 ko$=k1$+k8$+k2$+k8$+k3$+k8$+k
4$+k8$+k5$+k8$+k6$+k8$+k7$             <fi>
4310 rem rueckseiten                   <hm>
4320 k9$=l4$+c5$:fori=1to10:k9$=k9
$+chr$(166):nexti:k9$=k9$+chr$(5)+
l5$                                    <ib>
4330 kg$=k1$+k8$+k9$+k8$+k9$+k8$+k
9$+k8$+k9$+k8$+c9$+right$(str$(nr)
,1)+c5$+right$(k9$,12)+k8$+k7$         <jf>
4340 return                            <jg>
4350 rem kartendarstellung             <bn>
4360 rem *****************             <kn>
4370 restore 4680                      <eg>
4380 fori=1to18:readpw(i):nexti:re
turn                                   <fl>
4390 rem meldungen                     <cd>
4400 rem *********                     <ek>
4410 restore4710:fori=1to12:read m
$(i),m(1,i),m(2,i):nexti:return        <hm>
4420 rem fehlerbehandlung              <ng>
4430 rem ****************              <en>
4440 if er=30 then resume              <mc>
4450 slow:window0,0,39,24:scnclr:h
elp:printerr$(er)" error in "el:en
d                                      <on>
4460 rem titelbild-datas               <fa>
4470 rem ***************               <ni>
4480 data0,7,16,6,32,14,48,15,64,1
6,80,2,96,8,112,9,128,3,144,5,160,
7,176,6,192,14,208,15,224,16,240,2
,256,8,272,9,288,3,304,5               <cf>
4490 data0,5,16,3,32,9,48,8,64,2,8
0,16,96,15,112,14,128,6,144,7,160,
5,176,3,192,9,208,8,224,2,240,16,2
56,15,272,14,288,6,304,7               <ih>
4500 data16,5,32,3,48,9,64,8,80,2,
96,16,112,15,128,14,144,6,160,7        <lb>
4510 data16,7,32,6,48,14,64,15,80,
16,96,2,112,8,128,9,144,3,160,5        <el>
4520 datab,o,e,r,s,e,n,s,p,i,e,l,i
,n,c,o,m,p,u,t,e,r,v,e,r,s,i,o,n,(
,c,),s,t,e,p,h,a,n,s,c,h,a,e,l,i,1
,9,8,7                                 <kf>
4530 data2,8,9,3,5,7,6,14,15,16,12
,11                                    <je>
4540 data13,9,14,9,15,9,16,9,17,9,
18,9,19,9,20,9,21,9,22,9,23,9,24,9
,10,14,11,14,13,14,14,14,15,14,16,
14,17,14,18,14,19,14,20,14,21,14,2
2,14,23,14,24,14,25,14                 <po>
4550 data26,14,27,14,7,18,8,18,9,1
8,11,18,12,18,13,18,14,18,15,18,16
,18,17,18,19,18,20,18,21,18,22,18,
23,18,24,18,25,18,27,18,28,18,29,1
8,30,18                                <kk>
4560 rem spielfeld-datas               <nf>
4570 rem ***************               <fn>
4580 data 0,3,10,5,20,8,30,11,40,1
4,50,17,60,20,70,23,80,26,90,29,10
0,32,*,36                              <pn>
4590 data 35,58,82,106,130,154,178
,202,226,250,283                       <ff>
4600 data 59,78,97,116,135             <ng>
4610 data 44,94,64,94,89,94,112,94
,137,94,161,94,184,94,208,94,232,9
4,256,94,285,94,309,94                 <ek>
```




```
4620 data 44,113,64,113,89,113,112
,113,137,113,161,113,184,113,208,1
13,232,113,256,113,285,113,309,113 <hc>
4630 data 44,132,64,132,89,132,112
,132,137,132,161,132,184,132,208,1
32,232,132,256,132,285,132,309,132 <bb>
4640 data 44,151,64,151,89,151,112
,151,137,151,161,151,184,151,208,1
51,232,151,256,151,285,151,309,151 <ih>
4650 data 44,170,64,170,89,170,112
,170,137,170,161,170,184,170,208,1
70,232,170,256,170,285,170,309,170 <hp>
4660 rem kartendarstellung          <eh>
4670 rem *****************          <pf>
4680 data 1,0,14,0,27,0,1,8,14,8,2
7,8,1,16,14,16,27,16                <hl>
4690 rem meldungen                  <gp>
4700 rem *********                  <nh>
4710 data"forschungsprojekt erfolg
reich abge-     schlossen",1,4      <ph>
4720 data"auftragsbestand fuer neu
es jahr ge-     sichert",1,2        <jn>
4730 data"wegen schlechtem geschae
ftsgang wieder  500 arbeiter entla
ssen",-2,-4                         <ge>
4740 data"export ins ausland letzt
es jahr leicht  zurueckgegangen",0
,-2                                 <cc>
4750 data"allgemeine konjunkturerh
olung",0,2                          <gf>
4760 data"konjunkturflaute bahnt s
ich an",0,-2                        <co>
4770 data"steuerermaessigung fuer
unternehmer",1,2                    <dn>
4780 data"steuerbelastung fuer unt
ernehmer        groesser geworden"
,0,-2                               <kf>
4790 data"erdoelpreise sinken weit
erhin",0,4                          <bd>
4800 data"erdoelpreise sprunghaft
angestiegen",0,-3                   <hb>
4810 data"schwache dm beguenstigt
exporte",0,2                        <hj>
4820 data"starke dm schadet der ex
portwirtschaft",0,-2                <hj>
4830 rem ===================        <da>
```

Unsere Hotline: Jeden Mittwoch 15-19⁰⁰ Telefon 089/129 8013

Jetzt perfekt: Unser Checksummer

Hatte bisher unser Checksummer an Buchstabenvertauschungen nichts auszusetzen, so zeigt er sich nun nicht mehr so kulant.

Ob Sie mit der alten Version nun eingegeben hatten:

10 print"ab"
oder
10 print "ba",

der Checksummer brachte in beiden Fällen die Prüfsumme < gk > . Leicht kann es vorkommen, daß beim schnellen Tippen, besonders im Zehnfingersystem, die Taste, die eigentlich erst als übernächste drankommen sollte, ein wenig zu früh erwischt wird. Dem Checksummer, der lediglich die ASCII-Werte der Buchstaben addierte, konnte dieses natürlich nicht auffallen. Was also tun? Ob etwas früher oder später addiert wird, ändert nichts am Resultat der Summe. Anders ist es, wenn man zwei Verknüpfungsarten kombiniert. So ist z.B. 2*30+40 etwas anderes als 2*40+30. Und genau dieses war dann die Lösung. Die Summe wird nun einfach durch eine Linksverschiebung vor jeder Addition verdoppelt. Dadurch, daß im Falle, wenn das Ergebnis größer als 255 ist, der dabei entstehende Übertrag als Wert 1 zusätzlich addiert wird, verflüchtigen die Werte der am Anfang der Zeile gefundenen Codes sich nicht nach 8 weiteren Zeichen. Damit bleibt nicht nur die Aussagekraft der Prüfsumme voll erhalten, sondern erfäh- sogar eine erhebliche Steigerung. Und vor allen Dingen wird nur eine klitzekleine Änderung erforderlich, die dieses zu vollbringen in der Lage ist. Ein einziges Byte ist nur zu ändern. Wir tun dieses mit "poke 345,10" in der Zeile 470. Dadurch wird das hier ursprünglich ansässige CLC (Clear Carry) durch ASL (Arithmetik Shift Left) ersetzt. Die nachfolgende Addition mit ADC (Addiere mit Carry) addiert den ASCII-Code des gefundenen Zeichens und den nach links herausgeshifteten Übertrag. Da einige unserer Leser beklagten, daß das Checksummerlisting nachher noch im Programmspeicher stehen würde, haben wir diesem noch mit einem "new" abgeholfen. New bzw. neu ist nun folgendes.

10 print"ab" ergibt die Prüfsumme <jd>
10 print"ba" die Prüfsumme <jf>

Sie brauchen den Checksummer nicht neu einzutippen. Alles, was Sie tun müssen, ist, die Zeile 470 anzufügen. An der Bedienung des Checksummers hat sich nichts geändert. Die Eingabehinweise bleiben daher wie gehabt.

EINGABEHINWEISE

Am rechten Rand jedes Listings, jeweils am Ende einer Eingabezeile, finden Sie zwei Buchstaben zwischen einem Kleiner- und einem Größerzeichen eingeschlossen. Diese dürfen Sie nicht mit in Ihr Listing eintippen, sondern sie dienen Ihnen zur Überprüfung Ihrer Eingabe.

Zwischen dem Kleiner- und dem Größerzeichen am rechten Rand befinden sich zwei Buchstaben. Mit einem speziellen Programm können Sie beim Eintippen Ihre Eingabe auf ihre Richtigkeit überprüfen. Dieses Programm, der Checksummer, sorgt nämlich dafür, daß nach erfolgter Zeileneingabe am linken oberen Bildschirmeck zwei Buch-

ERST SICHERN, DANN AUSPROBIEREN

staben ausgegeben werden. Wenn diese Buchstaben nicht mit den vorher erwähnten Buchstaben in unserem Listing übereinstimmen, so können Sie davon ausgehen, daß Sie sich vertippt haben und können sich so die Zeile nochmals näher ansehen, ob Sie Ihren Eingabefehler finden. Wenn Sie dann alles richtig getippt haben, so stimmen die Buchstaben überein und Sie können sich getrost der nächsten Zeile zuwenden.

Das Checksummerlisting hat noch keine Prüfsummen. Seien Sie deshalb besonders aufmerksam, daß alles paßt und speichern Sie dieses Programm unbedingt ab, bevor Sie es starten! Bei einem Tippfehler würde es sich wahrscheinlich auf Nimmerwiedersehen verabschieden und Sie müßten die ganze Arbeit vermutlich nochmals machen. Wenn Sie es gestartet haben, so geschieht nichts Besonderes. Der Computer meldet sich einfach kurz darauf mit „READY", und das war auch schon alles. Alles sollte nun wie immer funktionieren, mit der kleinen Ausnahme, daß nunmehr nach jeder Eingabe im Direktmodus eine Prüfsumme erscheint. Nehmen Sie zum Testen irgendeine kurze BASIC-zeile aus unserem Heft her und testen sie aus. Wenn die Summen übereinstimmen, so können Sie sich freuen, denn Fehler beim Abtippen werden Ihnen nun in Zukunft viel weniger passieren, als vorher.

EINER FÜR ALLE, EIN ECHTES UNIVERSAL-PROGRAMM

Unseren Checksummer können Sie verwenden, ob Sie einen C16/116/Plus4 oder ob Sie einen C64 oder gar einen C128 haben. Nur müssen Sie beim letzteren beachten, ob Sie auch wirklich im 40-Zeichenmodus sind. Nachdem Sie den Checksummer geladen und gestartet haben, können Sie Ihr Basicprogramm eingeben wie gewohnt, Sie können es abspeichern, Sie können auch laden, Sie können Kürzel verwenden und ob Sie ein paar Leerzeichen mehr oder weniger verwenden, der Checksummer läßt sich dadurch nicht aus der Fassung bringen. Ein bißchen Vorsicht sollte man allerdings walten lassen, wenn man Programme eingetippt hat, in denen Peeks und Pokes vorkommen. Es wird zwar nicht besonders häufig vorkommen, aber es könnte bisweilen ge-



```
10 rem =checksummer==c16 c64 c128==
20 rem (p) 05/87 commodore welt   ==
30 rem ============================
40 rem (c) alfons mittelmeyer     ==
50 rem                            ==
60 rem c16/116/plus4              ==
70 rem c64                        ==
80 rem c128 (40-zeichen)          ==
90 rem ============================
100 rem ---------------------------
110 rem   grundroutine (c16)
120 rem ---------------------------
130 data165,059,072,165,060,072,032
140 data086,137,104,133,060,104,133
150 data059,152,072,160,000,165,020
160 data024,101,021,170,024,144,011
170 data201,032,240,006,138,024,113
180 data059,234,170,200,177,059,234
190 data208,240,169,031,072,138,074
200 data074,074,074,072,138,041,015
205 data072,169,031,072,162,003,104
210 data024,105,129,157,000,012,202
220 data016,246,104,168,096
230 lt=peek(772):ht=peek(773)
240 fori=312to386:readx:pokei,x:nex
t
250 iflt<>124then350
260 rem ---------------------------
270 rem        anpassung c64
280 rem ---------------------------
290 fori=312to317:pokei,234:next
300 fori=321to326:pokei,234:next
310 fori=1to6:readad:readx:pokead,x
:next
320 poke380,4:poke319,lt:poke320,ht
:goto430
330 data346,121,347,000,348,002
340 data351,185,352,000,353,002
350 iflt<>13then430
360 rem ---------------------------
370 rem anpassung c128 (40 zeichen)
380 rem ---------------------------
390 restore410:poke332,22
400 poke335,23:goto310
410 data313,061,316,062,323,062
420 data326,061,347,061,352,061
430 poke772,056:poke 773,1
440 rem ---------------------------
450 rem ergaenzung 10/87
460 rem ---------------------------
470 poke 345,10:new
480 rem ===========================
490 rem = fuer hefte cw 7/87 bis   =
500 rem = cw 9/87 sowie cw128 5/87=
510 rem = und c16 6/87 ist die     =
520 rem = poke-anweisung in zeile  =
530 rem = 470 wegzulassen          =
540 rem ===========================
```

schehen, daß nach dem Laufenlassen eines Programmes weder der Checksummer noch sonst etwas mehr funktioniert, auch wenn dies bisher ohne Checksummer nicht der Fall gewesen sein sollte. Also bitte sichern Sie in jedem Falle Ihre Programme, bevor Sie sie ausprobieren.

Ein paar Dinge sollten Sie noch wissen. Wir drucken in unseren Listings des öfteren Punkte statt Leerzeichen. Wenn Ihnen nun aber Leerzeichen besser gefallen, so liefert der Checksummer natürlich eine falsche Summe. Wenn Sie diese auf Richtigkeit überprüfen wollen, so können Sie dies tun, indem Sie sie zuerst einmal so wie im Heft abtippen, und nachher, nachdem Sie sie nachgeprüft haben, einfach wieder die Punkte durch Leerzeichen ersetzen.

A. Mittelmeyer

EINGABE-HILFE FÜR MASCHINENPROGRAMME

Checkmon 128

Dieses superkurze Programm ermöglicht es Ihnen, sogenannte Hex-Dump-Listings im MONITOR-Modus einzugeben und die eingetippten Zeilen auch gleich per eingeblendeter Checksumme zu überprüfen.

War ein Listing noch so lang, die Tipparbeit noch so schweißtreibend: Wenn das abgetippte Programm fehlerfrei läuft, sind alle Mühen vergessen.
Das haben Sie mit unserem hinlänglich bekannten Checksummer für BASIC-Listings sicher schon selbst erlebt.
Bei den sogenannten Hex-Dump-Listings, die Sie nach Aufruf der Anweisung MONITOR in den C128 eingegeben haben, gibt es im Normalfall keine Fehlermeldung, etwa SYNTAX ERROR IN 10. Meistens verabschiedet sich das Programm auf Nimmerwiedersehen. Auch wissen Sie nicht, ob das Programm nach dem Absturz und einem Betätigen des RESET-Knopfes noch intakt ist. Daher bleibt Ihnen nichts anderes übrig, als es zur Fehlersuche erneut zu laden. Erneuter Frust ist also vorprogrammiert. Was nottut, ist eine Eingabehilfe auch für Maschinenprogramme, eben diese Hex-Dump-Listings.

CHECKMON UND TEDMON

Tippen Sie zunächst das BASIC-Laderprogramm CHECKMON.128 mit Hilfe des Checksummers ab, speichern Sie es und starten Sie es mit RUN. Versetzen Sie dann den C128 in den MONITOR-Modus, entweder mit der F8-Taste oder durch Eingabe von MONITOR <RETURN>.
Sehen Sie sich jetzt beispielsweise den Speicherbereich ab Adresse $1000 mit M 1300 <RETURN> an. Sie bemerken bestimmt den Unterschied zum normalen Aussehen des Bildschirms: Rechts neben dem Doppelpunkt hinter den acht Hexadezimal-Bytes finden Sie nicht die gewohnte ASCII-Darstellung des Speicherinhalts, sondern eine in Spitzklammern eingekleidete Prüfsumme, wie Sie sie in ähnlicher Form schon von unserem Checksummer her kennen. Ändern Sie ein paar Speicherinhalte, indem Sie die angezeigten Hexzahlen durch andere, beliebige, überschreiben – und schon ändert sich auch die Prüfsumme.

DAS >-ZEICHEN

Das wichtigste Zeichen zur Eingabe von Hex-Dump-Zeilen ist die Spitzklammer nach rechts. Damit weiß der C128, daß nun nach der ebenfalls einzugebenden Adresse eine aneinandergereihte Eingabe von acht Hexadezimalwerten kommen muß, durch Leerzeichen getrennt. Verfahren Sie exakt so, wie Sie es bei einem Hex-Dump-Listing in diesem Heft abgedruckt sehen.
Nach Drücken der RETURN-Taste, also bei Übernahme dieser Maschinencode-Zeile, erscheint die Checksumme am Ende der eingetippten Zeile. Vergleichen Sie sie mit der im Heft. Müssen Sie verbessern, so fahren Sie mit dem Cursor wieder in diese Zeile hoch, ändern Sie die falschgeschriebenen Bytes und drücken erneut <RETURN>.
Ist jetzt alles in Ordnung, so beginnen Sie mit der Eingabe der nächsten Listingszeile wie gehabt.
Haben Sie alle Zeilen eines Hex-Dump-Listings fehlerfrei eingegeben, so dürfen Sie sicher sein, daß Sie das Programm so übernommen haben, wie es im Heft abgedruckt war.

SPEICHERN UND LADEN

Zum Abspeichern eines solchen Maschinen-Programms aus dem MONITOR-Modus heraus, benützen Sie die im Handbuch genannte Befehlseingabe:

S "(Prg.Name)" 08 Anfangsadresse Endadresse + 1

Den genauen Wortlaut finden Sie bei jeder Programmbeschreibung zu unseren Hex-Dump-Listings.
Wollen Sie ein Maschinensprache-Listing nicht auf einmal eingeben, so können Sie jederzeit bis zur aktuell eingegebenen Speicheradresse mit S speichern und das nächste Mal zum Weiterarbeiten mit einer anderen MONITOR-Anweisung wieder laden:

L "(Prg.Name)" 08

Am besten, Sie kreuzen sich im Listing aus dem Heft an, ab welcher Stelle oder Speicheradresse Sie Ihre weiteren Eingaben machen müssen.
Last not least: Es kann nie schaden, sich zu diesem Thema den Abschnitt C des Handbuches zum C128 über den eingebauten Maschinensprache-Monitor TEDMON durchzulesen.

Alfons Mittelmeyer/hb □

```
40 rem checkmon ===============128 <oj>
50 rem (c) cw 128 spezial      === <oc>
60 rem =========================== <nk>
70 rem by alfons mittelmeyer   === <eb>
80 rem c128                    === <al>
90 rem =========================== <jg>
100 fori=312to398:reada:pokei,a:ne
xt                                 <hc>
110 fori=1to4:reada:readb:pokea,b:
next:new                           <ef>
138 data 132,255,108,251,0,132,251
,164                               <co>
140 data 255,76,121,239,201,62,208
,249                               <dc>
148 data 165,102,10,101,103,160,7,
10                                 <pk>
150 data 113,102,136,16,250,133,25
3,169                              <pi>
158 data 30,133,254,169,62,160,97,
208                                <ge>
160 data 220,198,254,208,218,160,1
05,208                             <lk>
168 data 212,201,13,240,4,164,255,
24                                 <he>
170 data 96,169,60,160,68,32,61,1  <gl>
178 data 165,253,32,194,184,169,62
,160                               <me>
180 data 5,208,2,169,32,32,121,239 <kb>
188 data 136,208,248,169,13,208,17
6                                  <bi>
190 data251,68,252,1,806,56,807,1  <an>
```